



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB
PARA LA GESTIÓN ACADÉMICA Y FINANCIERA PARA EL
CENTRO DE CAPACITACIÓN ACADÉMICA Y
PREUNIVERSITARIA “GENIOS TRABAJANDO” MEDIANTE LA
UTILIZACIÓN DE LOS FRAMEWORKS PRIMEFACES/JPA

Trabajo de titulación presentado para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTORAS: MERCY LILIANA SANI CENTENO

PATRICIA ALEXANDRA ALLAUCA OÑATE

TUTOR: DR. JULIO SANTILLÁN CASTILLO

Riobamba – Ecuador

2016

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal de Trabajo de Titulación certifica que: El proyecto Técnico: **“DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN ACADÉMICA Y FINANCIERA PARA EL CENTRO DE CAPACITACIÓN ACADÉMICA Y PREUNIVERSITARIA “GENIOS TRABAJANDO” MEDIANTE LA UTILIZACIÓN DE LOS FRAMEWORKS PRIMEFACES/JPA**”, de responsabilidad de las señoritas Patricia Alexandra Alauca Oñate y Mercy Liliana Sani Centeno, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Dr. Miguel Tasambay, Ph. D. DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Dr. Julio Santillán DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
Dr. Julio Santillán DIRECTOR DEL TRABAJO DE TITULACIÓN	_____	_____
Ing. Raúl Rosero MIEMBRO DE TRIBUNAL	_____	_____

©2016, Mercy Liliana Sani Centeno, Patricia Alexandra Allauca Oñate

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el derecho de autor.

Nosotras, Mercy Liliana Sani Centeno y Patricia Alexandra Allauca Oñate, somos responsables de las ideas, métodos y las conclusiones obtenidas en el presente trabajo de titulación, y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo.

Mercy Liliana Sani Centeno

Patricia Alexandra Allauca Oñate

DEDICATORIA

El presente trabajo va dedicado a Dios primeramente por permitirme llegar hasta este punto y culminar mis objetivos. A mi esposo Javier Tenelema, a mis padres Walter Sani y Filomena Centeno y mis hermanas Alicia, Erika y Livia, quienes estuvieron presentes en cada momento siendo las principales fuentes de motivación y apoyo para poder concluir esta etapa. A ellos este proyecto porque sin ellos no lo hubiera podido terminar. Y a todas las personas, familia y amigos que confiaron en mí y me apoyaron hasta la culminación de mis estudios. Gracias y bendiciones.

Mercy Sani

El trabajo de titulación se la dedico principalmente a Dios por haberme dado la vida y haberme permitido llegar hasta este momento tan importante de mi formación profesional. A mis padres Luis y Eulalia por su cariño y apoyo incondicional. A mi familia, a mis hermanos Juan Carlos y Verónica por siempre animarme y motivarme a seguir adelante y cumplir con todos mis objetivos. A mi sobrinita Camila que es el motivo más grande que Dios me regalo para soportar los obstáculos que se presentaron en el trayecto de esta etapa.

Patricia Allauca

AGRADECIMIENTO

Agradezco a Dios por guiarme, otorgarme sabiduría y respaldo en todas las circunstancias a lo largo de mi vida estudiantil hasta el día de hoy, que culminó exitosamente esta etapa. A mi esposo, padres y hermanas por su apoyo incondicional a lo largo de mi vida estudiantil, lo cual me permite hoy alcanzar un sueño muy anhelado.

De igual forma extiendo mi agradecimiento al Dr. Julio Santillan, Ing. Raúl Rosero por sus consejos, apoyo y sus conocimientos en el desarrollo de este proyecto de titulación y al Ing. Diego Mayorga por hacer posible este proyecto. No quisiera culminar sin agradecer a mi amiga Patricia Allauca quien me ha brindado su amistad incondicional en los buenos y malos momentos.

Mercy Sani

Quiero agradecer a Dios ya que supo guiarme en cada una de las etapas de mi carrera, por darme fuerzas para seguir adelante y no permitirme desmayar con los problemas que me aquejaban. A mis maestros por haberme brindado los conocimientos necesarios para lograr alcanzar mi título. A mis padres les agradezco por haberme dado la vida, por su apoyo, sus consejos, su amor y sobre todo su paciencia, los amo con todo mi corazón gracias a ustedes soy lo que soy. Gracias hermanitos por su cariño y la confianza que siempre me la han tenido. Gracias a toda mi familia que siempre me han apoyado en los buenos y malos momentos.

Extiendo mi agradecimiento al Dr. Julio Santillán, Ing Raúl Rosero por haber contribuido con sus conocimientos para el desarrollo de este proyecto, al Ing. Diego Mayorga por su paciencia y su apoyo durante esta etapa y a Mercy por ser una excelente amiga y compañera, por haber realizado el trabajo de titulación puesto que hace aproximadamente 5 años quedamos en obtener nuestra ingeniería juntas.

Patricia Allauca

TABLA DE CONTENIDO

CERTIFICACIÓN.....	ii
DERECHO DE AUTOR	iii
DECLARACIÓN DE RESPONSABILIDAD.....	iv
DEDICATORIA.....	v
AGRADECIMIENTO	vi
TABLA DE CONTENIDO.....	vii
ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xii
RESUMEN.....	xiii
SUMMARY	xiv
INTRODUCCIÓN.....	1
OBJETIVOS.....	6
OBJETIVO GENERAL	6
OBJETIVOS ESPECÍFICOS	6
CAPÍTULO I	7
1 MARCO TEÓRICO REFERENCIAL.....	7
1.1 Aplicación web	7
1.1.1 <i>Definición</i>	7
1.1.2 <i>Estructura</i>	7
1.1.3 <i>Ventajas</i>	7
1.1.4 <i>Desventajas</i>	7
1.2 Modelo Vista Controlador	8
1.2.1 <i>Definición</i>	8
1.2.2 <i>Modelo</i>	8
1.2.3 <i>Vista</i>	8
1.2.4 <i>Controlador</i>	9
1.2.5 <i>Ciclo de Vida</i>	9
1.2.6 <i>Ventajas</i>	10
1.2.7 <i>Desventajas</i>	10
1.3 Frameworks	11
1.3.1 <i>Definición</i>	11
1.3.2 <i>Ventajas</i>	11
1.3.3 <i>Desventajas</i>	11

1.4	Frameworks de Persistencia	11
1.4.1	ORM	12
1.4.1.1	Definición	12
1.4.1.2	Características	12
1.4.1.3	Ventajas	12
1.4.1.4	Desventajas	12
1.5	Java Server Faces	12
1.5.1	Definición	12
1.5.2	Elementos	13
1.5.3	Características	13
1.5.4	Ciclo de Vida	13
1.5.5	Ventajas	14
1.5.6	Desventajas	14
1.6	Managed Beans	15
1.6.1	Definición	15
1.6.2	Alcance	15
1.7	Primefaces	16
1.7.1	Definición	16
1.7.2	Características	16
1.7.3	Componentes	16
1.7.4	Ventajas	21
1.7.5	Desventajas	21
1.8	JDBC	21
1.8.1	Definición	21
1.8.2	¿Qué hace JDBC?	21
1.8.3	Características	22
1.9	PostgreSQL	22
1.9.1	Definición	22
1.9.2	Arquitectura	22
1.9.3	Características	23
1.9.4	Límites	23
1.9.5	Ventajas	24
1.9.6	Desventajas	24
1.10	JPA	24
1.10.1	Definición	24
1.10.2	Arquitectura	25

1.10.3	Elementos	25
1.10.3.1	Entidades	25
1.10.3.1.1	Ciclo de Vida	26
1.10.3.1.2	Anotaciones	26
1.10.3.1.3	Relación entre entidades	26
1.10.3.2	Contexto de Persistencia	27
1.10.3.3	Unidad de Persistencia	27
1.10.3.4	Ventajas	29
1.10.3.5	Desventajas	29
1.11	Proveedor de Persistencia	30
1.11.1	EclipseLink	30
1.12	Ireport	30
1.12.1	Definición	30
1.12.2	Características	30
1.12.3	Funciones	31
CAPÍTULO II		32
2	MARCOMETODOLÓGICO	32
2.1	Diseño de la Investigación	32
2.2	Métodos	32
2.3	Técnicas	33
2.4	Metodología Scrum	33
2.4.1	Ciclo de Vida	33
2.4.2	Artefactos	34
2.4.3	Roles	34
2.5	Metodología Scrum para el desarrollo del Sistema	35
2.5.1	Planificación	35
2.5.1.1	Personas y Roles	35
2.5.1.2	Reuniones	35
2.5.1.3	Product Backlog	36
2.5.1.4	Plan de Entrega	38
2.5.2	Desarrollo	41
2.5.2.1	Sprint 1	41
2.5.2.1.1	Arquitectura del Sistema	41
2.5.2.1.1.1	Diagrama de Despliegue	41
2.5.2.1.1.2	Diagrama de Componentes	42
2.5.2.1.1.3	Diagrama de Casos de Usos	43

2.5.2.1.2	<i>Tecnologías de desarrollo</i>	43
2.5.2.1.3	<i>Estándar de Codificación</i>	44
2.5.2.1.4	<i>Diseño de la Base de Datos</i>	45
2.5.2.1.4.1	<i>Modelo de Datos Conceptual</i>	45
2.5.2.1.4.2	<i>Modelo de Datos Físico</i>	47
2.5.2.1.4.3	<i>Diccionario de Datos</i>	48
2.5.2.1.5	<i>Interfaz de Usuario</i>	48
2.5.2.2	<i>Sprint 2</i>	50
2.5.2.2.1	<i>Avance del Proyecto</i>	52
CAPÍTULO III		53
3	MARCO DE RESULTADOS Y DISCUSIÓN DE RESULTADOS	53
3.1	Técnicas utilizadas para el análisis y obtención de resultados	53
3.1.1	Observación	53
3.1.1.1	<i>Fichas de Observación con respecto al tiempo</i>	53
3.1.1.1.1	<i>Análisis Comparativo de la toma de Tiempos</i>	59
3.1.1.2	<i>Fichas de Observación con respecto a la cantidad de errores</i>	60
3.1.2	Encuesta	65
3.1.2.1	<i>Validación de la usabilidad del sistema “SGAF”</i>	71
3.2	Análisis de Resultados	72
CONCLUSIONES		74
RECOMENDACIONES		75
GLOSARIO		76
BIBLIOGRAFÍA		77

ÍNDICE DE TABLAS

Tabla 1-1:	Etiquetas/componentes de Ajax Core	16
Tabla 2-1:	Etiquetas/componentes de Input	17
Tabla 3-1:	Etiquetas/componentes de Button.....	17
Tabla 4-1:	Etiquetas/componentes de Data	17
Tabla 5-1:	Etiquetas/componentes de Panel.....	18
Tabla 6-1:	Etiquetas/componentes de Overlay	18
Tabla 7-1:	Etiquetas/componentes de Menu	19
Tabla 8-1:	Etiquetas/componentes de Charts	19
Tabla 9-1:	Etiquetas/componentes de Message.....	19
Tabla 10-1:	Etiquetas/componentes de Multimedia	20
Tabla 11-1:	Etiquetas/componentes de Files	20
Tabla 12-1:	Etiquetas/componentes de DragDrop.....	20
Tabla 13-1:	Etiquetas/componentes de Misc.....	20
Tabla 14-1:	Límites PostgreSQL	24
Tabla 15-1:	Anotaciones	26
Tabla 16-1:	Métodos Entity Manager.....	29
Tabla 1-2:	Tareas Realizadas.....	35
Tabla 2-2:	Recurso Humano - Roles	35
Tabla 3-2:	Product Backlog.....	37
Tabla 4-2:	Requerimientos del Sistema.....	39
Tabla 5-2:	Arquitectura del Sistema.....	41
Tabla 6-2:	Tecnologías de Desarrollo	43
Tabla 7-2:	Estándar de Codificación	44
Tabla 8-2:	Diseño de la BD.....	45
Tabla 9-2:	Interfaz de Usuario.....	48
Tabla 10-2:	Historia de Usuario HU11.....	50
Tabla 11-2:	Tarea de Ingeniería 1 HU11.....	51
Tabla 12-2:	Prueba de Aceptación 1 HU11	51
Tabla 1-3:	Ficha de Observación (Tiempo sin Sistema)	54
Tabla 2-3:	Ficha de Observación (Tiempo con Sistema)	55
Tabla 3-3:	Proceso 1	56
Tabla 4-3:	Proceso 2.....	56
Tabla 5-3:	Proceso 3.....	57
Tabla 6-3:	Proceso 4.....	58
Tabla 7-3:	Proceso 5.....	58
Tabla 8-3:	Análisis Comparativo Tiempo	59
Tabla 9-3:	Ficha de Observación (Cantidad de errores encontrados sin el sistema)	61
Tabla 10-3:	Procesos realizados sin la utilización del sistema	62
Tabla 11-3:	Número de errores por cada proceso.....	63
Tabla 12-3:	Ficha de Observación (Cantidad de errores encontrados con el sistema)	64
Tabla 13-3:	Personas Encuestadas.....	65
Tabla 14-3:	Pregunta 1	66
Tabla 15-3:	Pregunta 2	66
Tabla 16-3:	Pregunta 3	67
Tabla 17-3:	Pregunta 4	68
Tabla 18-3:	Pregunta 5	68
Tabla 19-3:	Pregunta 6	69
Tabla 20-3:	Pregunta 7	70
Tabla 21-3:	Test de Usabilidad.....	71

ÍNDICE DE FIGURAS

Figura 1-1.	Ciclo de Vida MVC	10
Figura 2-1.	Ciclo de Vida JSF	14
Figura 3-1.	Alcance MB	15
Figura 4-1.	Arquitectura de PostgreSQL	22
Figura 5-1.	Arquitectura Externa JPA	25
Figura 6-1.	Contexto de Persistencia	27
Figura 7-1.	Archivo persistence.xml	28
Figura 1-2.	Ciclo de Vida SCRUM	33
Figura 2-2.	Roles SCRUM	34
Figura 3-2.	Arquitectura del Sistema	42
Figura 4-2.	Diagrama de Componentes	42
Figura 5-2.	Diagrama de Caso de Uso	43
Figura 6-2.	Modelo de Datos Conceptual	46
Figura 7-2.	Modelo de Datos Físico	47
Figura 8-2.	Login SGAF	49
Figura 9-2.	Página Principal	49
Figura 10-2.	Recibo Pago Docente	50
Figura 11-2.	BurnDownChart SGAF	52
Figura 1-3.	Resultados Proceso 1	56
Figura 2-3.	Resultados Proceso 2	57
Figura 3-3.	Resultados Proceso 3	57
Figura 4-3.	Resultados Proceso 4	58
Figura 5-3.	Resultados Proceso 5	59
Figura 6-3.	Procesos realizados sin la utilización del sistema	63
Figura 7-3.	Resultados Pregunta 1	66
Figura 8-3.	Resultados Pregunta 2	67
Figura 9-3.	Resultados Pregunta 3	67
Figura 10-3.	Resultados Pregunta 4	68
Figura 11-3.	Resultados Pregunta 5	69
Figura 12-3.	Resultados Pregunta 6	69
Figura 13-3.	Resultados Pregunta 7	70
Figura 14-3.	Resultados Usabilidad	72
Figura 1-A.	Información Aulas	80
Figura 2-A.	Información Cursos	80
Figura 3-A.	Información Docentes	81
Figura 4-A.	Información Inscripciones y Cuotas	81
Figura 5-A.	Información Proyección y Cálculo Pago	82
Figura 6-A.	Información Pago a Docentes	82
Figura 1-C.	Planificación del Trabajo de Titulación “SGAF”	121

RESUMEN

El trabajo de titulación se enfoca a implementar el Sistema de Gestión Académica y Financiera (SGAF) para el Centro de Capacitación Académica y Preuniversitaria GENIOS TRABAJANDO ubicado en la ciudad de Riobamba. Los métodos de investigación empleados fueron el análisis y síntesis que ayudaron a determinar el problema relacionado con la pérdida de tiempo en la atención y realización de cálculos, por la falta de información oportuna de parte de la secretaria. Las técnicas de recopilación de información utilizadas fueron la observación para obtener los tiempos empleados antes y después de utilizar el sistema, la encuesta para medir la usabilidad se la realizó a los 2 directivos y 3 secretarias. Para la implementación de la aplicación se utilizó software libre como el framework JPA, Primefaces, IDE Netbeans 8.0, gestor de base de datos PostgreSQL 9.4, JasperReport 7.4 para la generación de reportes, estos permiten desarrollar proyectos a nivel empresarial. La metodología de desarrollo de software SCRUM ayudó a cumplir cada uno de los requerimientos permitiendo la interacción del usuario durante el desarrollo del trabajo de titulación. Según el análisis realizado se determinó que se redujo el tiempo de respuesta en un total de 43 minutos con 20 segundos en la ejecución de los principales procesos como son inscripciones, cobros, rol de pagos, emisión de recibos y reportes. Los errores encontrados que correspondían al 75% de los procesos académicos y financieros que se llevaban manualmente, fueron solucionados en un 100%. En el análisis de la usabilidad se determinó que el sistema es aceptable en un 91.43%. Se concluye que el sistema consiguió reducir el tiempo de respuesta y dar solución a los errores encontrados, permitiendo obtener información oportuna en tiempo real, logrando rapidez en la realización de los procesos y la toma de decisiones por parte de los directivos. Recomendamos utilizar el sistema en la institución.

Palabras claves:

<SISTEMA INFORMÁTICO>, <FRAMEWORK DE PERSISTENCIA [JPA]>, <FRAMEWORK INTERFAZ DE USUARIO [PRIMEFACES]>, <LENGUAJE DE PROGRAMACIÓN [JAVA]>, <SISTEMA DE GESTIÓN ACADÉMICA Y FINANCIERA [SGAF]>, <METODOLOGÍA DE DESARROLLO [SCRUM]>, <SOFTWARE NETBEANS>

SUMMARY

The purpose of the present research is to implement the academic financial management system (AFMS) for Academic and Pre-University training Center “GENIOS TRABAJANDO”, located in the city of Riobamba. The research methods used were the analysis and synthesis which helped to determine the problem related to waste of time in the performing calculations, due to the lack of timely information from the secretariat. On the other hand, the data collection techniques applied were observation to obtain the times used before and after using the system, besides the survey technique was applied to two managers and three secretaries to measure its usability. For its implementation used free software such as: framework JPA, Primefaces, Netbeans IDE 8.0, database manager Postgres SQL 9.4, Jasper Report 7.4 for generating reports that allow developing projects according to enterprise level. The development methodology of SCRUM software helped to reach each one of the requirements, allowing to user interacts during the work degree. According to the analysis, it determined that the response time was reduced by a total of 43 minutes with 20 seconds on the implementation of main processes such as: registration, charges, payments, issuance of receipts and reports. At the same time, errors identified and correspond to 75% of academic and financial processes carried out manually were solved by 100%. The usability analysis determined that the system is acceptable in a 91.43%. It concludes that the system was able to reduce the response time and resolve errors found, obtaining timely information in real time, doing the processes faster and decision-making by managers. Finally, it recommends using the system at the institution.

Keywords:

<COMPUTER SYSTEM>, <FRAMEWORK OF PERSISTENCE [JPA]>, <USER INTERFACE FRAMEWORK [PRIMEFACES]>, <PROGRAMMING LANGUAGE [JAVA]>, <ACADEMIC FINANCIAL MANAGEMENT SYSTEM [AFMS]>, <MEHODOLOGY OF DEVELOPMENT [SCRUM]>, <SOFTWARE NETBEANS>

INTRODUCCIÓN

En la actualidad las empresas u organizaciones optan por optimizar los procesos que se llevan a cabo para lo cual es necesario contar con sistemas potentes que permitan automatizar la información.

Con el desarrollo de aplicaciones web se ha logrado obtener un avance significativo, permitiendo desarrollar sistemas o aplicaciones automatizadas que puedan efectuar las tareas que se realizan y así mejorar la atención, facilidad y agilidad en los procesos, todo esto mediante la utilización de nuevas herramientas o tecnologías las cuales han surgido con el pasar del tiempo.

Por lo cual se ha propuesto al “Centro de Capacitación Académica y Preuniversitaria GENIOS TRABAJANDO”, automatizar la información relacionada con la parte académica y financiera, que constituyen los procesos esenciales que son llevados a cabo a diario por parte de los empleados (secretarias) y los directivos lo que permitirá gestionar adecuadamente toda la información.

Este trabajo de titulación se ha dividido en tres capítulos donde el primero corresponde al marco teórico referencial en el cual se detalla los conceptos, características y demás aspectos relacionados a los frameworks de persistencia como JPA y herramientas para mejorar el diseño de las interfaces como es PrimeFaces, todo esto servirá como punto de partida para el desarrollo de la aplicación web.

El segundo capítulo corresponde al marco metodológico donde se detallará los métodos, técnicas y la metodología SCRUM de desarrollo de software, las mismas que servirán como base para desarrollar la aplicación web.

Por último en el tercer capítulo se detalla el análisis e interpretación de los resultados obtenidos con el desarrollo del Sistema web para la Gestión Académica y Financiera para el Centro de Capacitación Académica y Preuniversitaria “GENIOS TRABAJANDO”.

Antecedentes

Actualmente se requiere la automatización de la información que una empresa u organización maneja, por ende el desarrollo de aplicaciones web permiten que los usuarios accedan a la

información desde cualquier lugar de manera rápida evitando así pérdida de tiempo y demora en la realización de un proceso determinado.

Cuando se requiere desarrollar una aplicación en Java uno de los principales aspectos que se deben tener en cuenta es la integración con una base de datos, la misma que permitirá almacenar, actualizar y recuperar la información que utilice la aplicación, debido a que los datos de las empresas u organizaciones deben estar cuidadosamente protegidos.

Existen diversos mecanismos de persistencia que pueden ser utilizados, sin embargo esto dependerá de los desarrolladores, la persistencia constituye una de las necesidades básicas de cualquier sistema de información de cualquier tipo especialmente para el desarrollo de aplicaciones web, puesto que permitirá mapear objetos para optimizar velocidad y eficiencia en la base de datos.

Mediante los frameworks de persistencia se puede obtener mayor rendimiento en una aplicación web, ya que ayudan a solucionar los problemas de mapeo de objetos desde una base de datos relacional y viceversa, así como agilizar el tiempo al realizar las transacciones y acceso a la base de datos.

En el artículo publicado por Cecilio Álvarez afirma que JPA es el estándar de Java encargado de automatizar dentro de lo posible la persistencia de los objetos en la base de datos y estas herramientas se utilizan principalmente porque son libres, permiten ahorrar costos de implementación y pagos de licencias. (Caules, 2013)

Los frameworks utilizados para la creación de interfaces gráficas están compuestos por componentes personalizables y potentes que mejoran el desarrollo de las aplicaciones web de forma ágil y sencilla permitiendo ahorro de recursos y tiempo como es el caso de Java Server Faces que reduce el diseño y el desarrollo de interfaces al utilizar un conjunto de librerías formadas por componentes visuales como es Primefaces.

“GENIOS TRABAJANDO” es un centro educativo cuya visión es lograr que todos los estudiantes se preparen superando sus destrezas con capacidades para cualquier prueba en las diferentes instituciones que ellos aspiren ingresar. Empezó a llevar a cabo sus actividades el 5 de Noviembre del 2008.

El problema que se presenta en el Centro de Capacitación Académica y Preuniversitaria “GENIOS TRABAJANDO” es la pérdida de tiempo, demora en la atención, falta de información oportuna, preocupación por realizar cálculos del dinero diario y mensual que ingresa a dicha institución, provocando posibles errores o confusión al momento de realizar

algún cálculo, peor aún podría ser objeto de robo o pérdida de información valiosa, todo esto se da puesto que llevan un registro manual tanto de la parte académica como por ejemplo registro de estudiantes, docentes, capacitaciones, inscripciones, cursos así como la parte financiera como los cobros a los estudiantes, los pagos a los docentes, etc.

Por todo lo mencionado anteriormente se ha encontrado en la actualidad que no se cuenta con un sistema automatizado, lo que ha traído consigo conflictos en dicha institución se sugiere que se desarrolle una aplicación que permita automatizar los procesos.

Justificación del trabajo de titulación

Se detalla a continuación las razones tanto de la forma teórica como aplicativa del porqué se ha seleccionado el tema, dando a conocer los aspectos más importantes de las tecnologías a utilizar, siendo todo esto importante para el desarrollo del trabajo de titulación

Justificación Teórica

Se optó por utilizar las siguientes herramientas: el lenguaje de programación Java con el IDE Netbeans porque es uno de los más utilizados y con excelente documentación, el framework de persistencia JPA que proporciona la reducción en el código de acceso a la base de datos y permite el mapeo de las tablas a entidades. Primefaces porque proporciona gran cantidad de componentes que permiten crear interfaces entendibles para que el usuario pueda interactuar con facilidad, PostgreSQL es muy sencillo de manejar, nos ofrece seguridad, confiabilidad, estabilidad y la oportunidad de mejorar y aportar desarrollos funcionales para hacer un manejador de base de datos aún más óptimo.

Primefaces es un framework open source para Java Server Faces que cuenta con más de 100 componentes, está integrado con CSS, permite el soporte Ajax que es transparente para el desarrollador así como el uso de JavaScript y utiliza el soporte de jQuery para los efectos visuales y así facilitar la creación de las aplicaciones web contando para ello con una excelente documentación. (Cruz Farfán, 2012)

JPA es una especificación de Sun Microsystems Java encargado de automatizar dentro de lo posible la persistencia de los objetos JAVA en cualquier base de datos relacional, facilita la persistencia desde el punto de vista del desarrollador, no requiere un servidor de aplicaciones. (Rengifo, 2011)

PotgreSQL es un potente sistema de base de datos objeto-relacional, cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos. (Moycas, 2011)

Justificación Aplicativa

La razón para el desarrollo de la aplicación es que permita automatizar los procesos tanto académicos como financieros, que ayuden a reducir los tiempos de respuesta, los errores encontrados mejorando así la atención, el servicio e imagen de dicha institución puesto que es necesario contar con medios tecnológicos que agilicen los procesos que se ejecuten en el mismo, provocando que la información manejada por la institución sea administrada de una manera adecuada, esto ayudará a la organización, planeación y la toma de decisiones por parte de los directivos de la institución.

El sistema será desplegado en la web, el mismo que está enfocado a los siguientes módulos, en los que se detalla cada uno de los procesos a realizar:

- Módulo Académico: permitirá llevar a cabo los siguientes procesos:

Proceso de Registro de Datos.- no tienen un registro automatizado de los datos de los estudiantes, cursos, capacitaciones, docentes, horarios, períodos, institución teniendo en cuadernos o en hojas de Excel.

Proceso de inscripción.- este proceso se lo lleva de forma manual, los estudiantes escogen el horario según su comodidad, para realizar este proceso se cuenta con aulas las mismas que tienen su propia capacidad. El problema surge cuando durante las inscripciones no se sabe si existen aún cupos disponibles para dicho curso. No obstante el estudiante puede cambiar de horario obligando a la secretaria a modificar los registros.

Proceso de Modificación de Datos.- se lo realizará cuando haya que actualizar información que se requiera.

- Módulo Financiero: permitirá llevar a cabo los siguientes procesos:

Proceso para el Pago de la cuota del curso por parte del estudiante se refiere a que el estudiante debe de pagar por la inscripción un valor de 30\$, durante el transcurso del curso se debe cubrir el valor total. Debido a que los estudiantes pagan diariamente la secretaria debe llevar la contabilidad para posteriormente registrar en sus cuadernos.

Proceso para el Pago a los docentes que dictan las capacitaciones como Senescyt, Ineval, Genios (Física, Química, Matemáticas entre otras).

Proceso para el cálculo de la contabilidad diaria del dinero que se recaude tanto de las inscripciones y de los pagos que realizan los estudiantes, así como también la contabilidad mensual de lo que la institución ha obtenido y de esta manera pagar los servicios a los docentes y conocer cuáles son los ingresos y los egresos que la institución posee.

Proceso para la generación de recibos de las inscripciones y pagos que realizan los estudiantes, así como el rol de pagos para los docentes serán visualizados en archivos pdf.

Proceso de Generación de Reportes.- se realizará cuando sean requeridos por parte de los directivos de la institución.

OBJETIVOS

OBJETIVO GENERAL

Diseñar e implementar una aplicación web mediante los frameworks PRIMEFACES/JPA para el Centro de Capacitación Académica y Preuniversitaria “GENIOS TRABAJANDO”

OBJETIVOS ESPECÍFICOS

- Analizar los frameworks PRIMEFACES y JPA para el desarrollo de aplicaciones web empresariales.
- Desarrollar el sistema web para la Gestión Académica y Financiera del Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”
- Realizar las pruebas respectivas al sistema para determinar las mejoras encontradas por parte de los usuarios.

CAPÍTULO I

1 MARCO TEÓRICO REFERENCIAL

1.1 Aplicación web

1.1.1 Definición

Es cualquier herramienta que se encuentra dentro de un servidor web al que un usuario puede acceder mediante un navegador web y conexión a internet. (Uywork, 2015)

1.1.2 Estructura

- **Primera capa:** está constituida por un navegador web.
- **Segunda capa:** es la capa intermedia se compone por un motor que permite leer una tecnología web de tipo dinámico.
- **Tercera capa:** se integra por una base de datos sólida. (Uywork, 2015)

1.1.3 Ventajas

- ✓ No necesitan ser instaladas y para desplegar una aplicación web en las grandes organizaciones todo lo que se necesita es un navegador web compatible.
- ✓ No requieren actualizar los procedimientos puesto que todas las funciones están implementadas en el servidor y son enviadas automáticamente a los usuarios.
- ✓ Proporcionan compatibilidad entre plataformas (Windows, Linux, Mac) debido a que operan dentro de una ventana del navegador web, es decir no necesitan un sistema operativo en específico puesto que las aplicaciones están hechas para que sirvan en cualquier navegador. (Rojas, 2011)

1.1.4 Desventajas

- × Se necesita internet, debido a esto muchos usuarios o personas piensan y prefieren no utilizar estos servicios, puesto que no todos tienen internet y esto a la larga sería un problema muy grande.
- × La información de nuestra empresa, datos personales, documentos o archivos están almacenados en una determinada aplicación pero estos quedarían fácilmente expuestos hacer alterados o borrados puesto que no se tiene el control.
- × Dependencia a plugins como es el caso de Google que puede necesitar Flash, Java y otros plugins que la aplicación requiera. (Rojas, 2011)

1.2 Modelo Vista Controlador

Actualmente en cualquier parte del mundo, los desarrolladores centran la atención principalmente en cómo utilizar los estándares para el diseño de aplicaciones que permitan reutilizar el código, así como lograr la implementación de las aplicaciones en el menor tiempo posible.

MVC surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de la estandarización en el diseño de las aplicaciones. (Romero & Díaz, 2012)

1.2.1 Definición

MVC es una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario, surge de la necesidad de crear software más robusto con un ciclo de vida adecuado, donde se potencie la facilidad de mantenimiento, separación de conceptos y reutilización del código. (Álvarez, 2014)

Se trata de un modelo que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones así como en múltiples lenguajes de programación y plataformas de desarrollo. (Alicante, 2015)

Este patrón separa el código en tres capas que se detallan a continuación:

1.2.2 Modelo

Es la representación específica de la información con la que el sistema opera por lo tanto gestiona todos los accesos y la integridad de los datos, tanto consultas como actualizaciones, además de implementar los privilegios de acceso que se hayan descrito en las especificaciones de la lógica de negocio.

El Modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos, lo ideal sería que el modelo sea independiente del sistema de almacenamiento.
- Definir las reglas de negocio o funcionalidad del sistema.
- Llevar un registro de las vistas y los controladores del sistema. (Lozu, 2013)

1.2.3 Vista

Es la presentación de la información y lógica de negocio (modelo) que debe visualizarse como salida en un formato adecuado que permita interactuar con el usuario.

Las vistas son responsables de:

- Recibir datos del modelo y mostrarlos al usuario.

- Tener un registro del controlador asociado.
- Proporcionar el servicio de actualización, para que sea invocado por el controlador o por el modelo. (Alicante, 2015)

1.2.4 Controlador

Es la capa que actúa como intermediario entre el Modelo y la Vista, controla todo lo que puede realizar la aplicación, se encarga de responder a las diferentes acciones del usuario, de invocar peticiones al modelo cuando se hace alguna solicitud sobre la información, así como también envía comandos a la vista, ejecutando la acción adecuada y creando el modelo pertinente.

El Controlador es el responsable de:

- Recibir los eventos de entrada como un cambio en un campo de texto, un clic, etc.
- Contener reglas de gestión de eventos.

El patrón de arquitectura de software se encarga de separar los datos de una aplicación por ejemplo en las aplicaciones web la vista es la página HTML, el modelo es el SGBD y la lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista. (Alicante, 2015)

1.2.5 Ciclo de Vida

El ciclo de vida de MVC es normalmente representado por las 3 capas que son Modelo, Vista y Controlador, se los detalla a continuación:

- El usuario hace la solicitud al controlador con la acción que requiere realizar.
- El controlador decide delegar la tarea al modelo y éste empieza a realizar su trabajo.
- El modelo se encarga de realizar la tarea que le solicita el controlador, después le regresa la información resultante y a la vez redirige a la vista.
- La vista se encarga de transformar los datos en información entendible para el usuario.
- Finalmente, la representación gráfica es enviada de regreso al controlador y éste se encarga de transmitírsela al usuario. (Lozu, 2013)

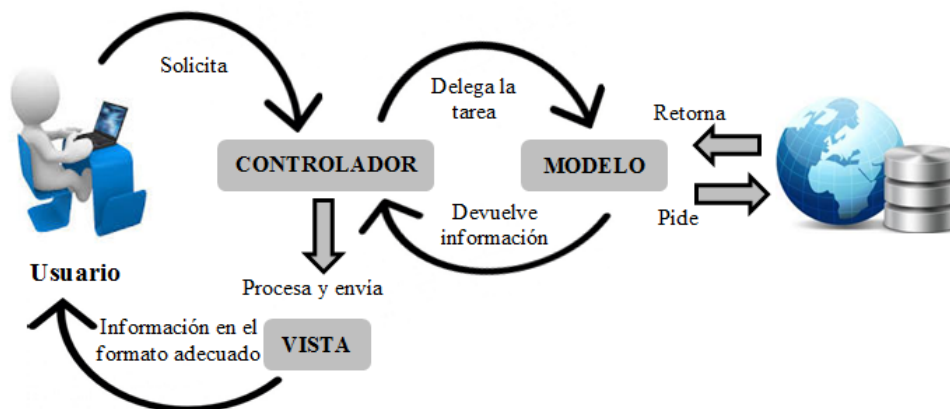


Figura 1-1. Ciclo de vida MVC

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.2.6 Ventajas

Al existir la separación de vistas, controladores y modelos es más sencillo realizar labores de mejora como:

- ✓ La aplicación está implementada modularmente.
- ✓ Las vistas muestran siempre información actualizada.
- ✓ Permite la separación entre la interfaz, lógica de negocio y la de presentación.
- ✓ Facilidad para la realización de pruebas unitarias de los componentes, o aplicar el desarrollo guiado por pruebas o TDD.
- ✓ Reutilización de los componentes.
- ✓ Agregar nuevas vistas, nuevas formas de recolectar las órdenes del usuario.
- ✓ Facilidad de modificar los objetos de negocios para migrar a otra tecnología.
- ✓ Las labores de mantenimiento se simplifican, las correcciones se debe realizar en un lugar específico.
- ✓ Las vistas también son susceptibles de modificación sin necesidad de provocar que todo el sistema se paralice.
- ✓ Facilidad para desarrollar prototipos así como el desarrollo de la aplicación es mucho más escalable. (Lozu, 2013)

1.2.7 Desventajas

- × Tiene que limitarse a una estructura predefinida, lo que a veces puede incrementar la complejidad del sistema.
- × El tiempo de desarrollo de una aplicación que implementa el patrón de diseño MVC es mayor en la primera parte del desarrollo puesto que se implementa una mayor cantidad de clases.
- × La curva de aprendizaje para los nuevos desarrolladores se estima mayor que la de modelos más simples como WebForms.

- × La distribución de componentes obliga a crear y mantener un mayor número de ficheros.
- × MVC es un patrón de diseño orientado a objetos por lo que su implementación es sumamente costoso y difícil en lenguajes que no siguen este paradigma.(Lozu, 2013)

1.3 Frameworks

1.3.1 Definición

Es un entorno o ambiente de trabajo que tiene una estructura de soporte definida, en la cual un proyecto de software puede ser organizado y desarrollado. Los frameworks suelen incluir:

- Soporte de programas
- Bibliotecas
- Scripting. (Climent, 2013)

1.3.2 Ventajas

- ✓ El programador no necesita plantearse una estructura global de la aplicación, sino que el framework le proporciona un patrón que hay que “rellenar”.
- ✓ Facilita la colaboración, es decir que cualquier programador que haya tenido que pelearse con el código fuente de otro o incluso con el propio, sabrá lo difícil que es entenderlo y modificarlo, para esto todo lo que sea definir y estandarizar va a ahorrar tiempo y trabajo a los desarrolladores.
- ✓ Proporcionan soluciones prefabricadas para los problemas más comunes, permitiendo a los desarrolladores centrarse en las necesidades de negocio es decir concentrar más esfuerzo y tiempo en identificar los requerimientos del software y así evitar resolver los detalles técnicos de bajo nivel.
- ✓ Ofrece una estructura clara y organizada a varios niveles, es decir es más fácil encontrar cualquier recurso cuando sea necesario cambiarlo. (Climent, 2013)

1.3.3 Desventajas

- × Cada framework tiene gran variedad de componentes que el desarrollador debe aprender.
- × Los frameworks consumen más recursos, debido a que la característica de reutilización tiende a generalizar la funcionalidad de los componentes. El resultado es que se incluyen características que están de más, provocando una sobrecarga de recursos.
- × No todos los frameworks son conocidos.(Climent, 2013)

1.4 Frameworks de Persistencia

Permite el almacenamiento de la información después de la finalización del programa, puesto que la mayoría de los sistemas actuales necesitan preservar los datos para su posterior uso.

Existen varios frameworks de persistencia como Hibernate, Ibatis, JPA entre otros.

1.4.1 ORM

1.4.1.1 Definición

Este modelo de programación consiste en la transformación de las tablas de una base de datos en una serie de entidades que simplifiquen las tareas básicas de acceso a los datos para el desarrollador. (TuProgramación, 2013)

1.4.1.2 Características

- El ORM al tener una capa intermedia, abstrae al programador de la base de datos y centra su atención únicamente en el desarrollo de la aplicación.
- Facilita las labores básicas de cualquier acceso a datos como son las operaciones CRUD (Create, Read, Update y Delete) a través de un lenguaje de alto nivel orientado a objetos. (TuProgramación, 2013)

1.4.1.3 Ventajas

- ✓ Posee facilidad y velocidad de uso.
- ✓ Permite la abstracción de la base de datos usada.
- ✓ Proporciona seguridad de la capa de acceso a datos contra ataques.
- ✓ Reutilización, esto permite llamar a los métodos de un objeto de datos. (TuProgramación, 2013)

1.4.1.4 Desventajas

- × Curva de aprendizaje alta en un nuevo lenguaje ORM.
- × En entornos con gran carga poner una capa más en el proceso puede reducir el rendimiento. (TuProgramación, 2013)

1.5 Java Server Faces

Los desarrolladores están utilizando tecnologías como Servlets y JSP para desarrollar interfaces de usuario basadas en navegadores escalables y robustos para un sinnúmero de aplicaciones con gran éxito.

Pero a medida que las aplicaciones web se hacen más complejas, algunos desarrolladores requieren optar por nuevas, mejores y agradables interfaces gráficas de usuario. (Peña, 2013)

1.5.1 Definición

Es un framework que simplifica el desarrollo sofisticado de aplicaciones web destinadas a facilitar la integración del desarrollo de interfaces de usuario en aplicaciones Java EE, en la que podemos crear interfaces de usuario sencilla, con un conjunto de componentes reutilizables.(Peña, 2013)

Se basa en el API de Servlets que proporciona un conjunto de componentes en forma de etiquetas definidas en las páginas XHTML mediante el framework Facelets el mismo que constituye un elemento fundamental que proporciona características de plantillas y de creación de componentes compuestos.(Alicante, 2014)

1.5.2 Elementos

Los elementos que se incluyen en una aplicación de Java Server Faces son:

- Un conjunto de páginas JSP.
- Un conjunto de beans que definen propiedades y funciones para los componentes de la interfaz de usuario.
- Un archivo de configuración de los recursos de la aplicación.
- Un descriptor de despliegue.
- Un conjunto de objetos personalizados desarrollados por el programador.
- Un conjunto de etiquetas para trabajar con objetos personalizados. (González, 2013)

1.5.3 Características

- Permite la definición de las interfaces de usuario mediante vistas que agrupan componentes gráficos.
- Asocia a cada vista con formularios un conjunto de objetos java manejados por el controlador Managed Bean que facilitan la recogida, manipulación y visualización de los valores mostrados en los diferentes elementos de los formularios.
- Introduce una serie de etapas en el procesamiento de la petición, como por ejemplo la de validación, reconstrucción de la vista, recuperación de los valores de los elementos, etc.
- Utiliza un fichero sencillo de configuración para el controlador en formato xml.
- Es extensible, pudiendo crearse nuevos elementos de la interfaz o modificar los ya existentes.(Muñoz, 2012)

1.5.4 Ciclo de Vida

El ciclo de vida no es más que una secuencia de fases por las que pasa una petición JSF desde que se recibe en el servidor hasta que se genera la página HTML resultante.

Las fases del ciclo de vida son las siguientes:

1. Restaurar la vista (*restore view*).- se obtiene el árbol de componentes correspondientes a la vista JSF de la petición.
2. Aplicar los valores de la petición (*apply request value*).- se procesa todos los valores asociados a los componentes convirtiendo los datos de la petición a tipo de datos Java.
3. Procesar las validaciones (*process validations*).- se validan todos los datos, si existe algún error se encola un mensaje de error y se termina el ciclo de vida, saltando al último paso (renderizar la respuesta).
4. Actualizar los valores del modelo (*update model values*).- todos los valores se han procesado y se han validado, además se actualizan las propiedades de los beans gestionados asociados a los componentes.
5. Invocar la aplicación (*invoke application*).- todas las propiedades de los beans asociados a los componentes de entrada se han actualizado.
6. Renderizar la respuesta (*render response*).- se muestra la respuesta al usuario.(Argenis, 2013)

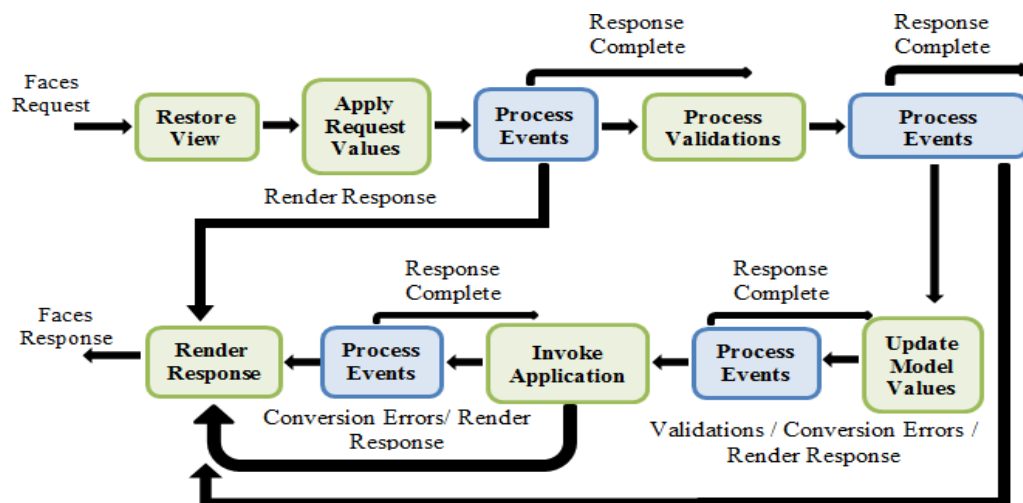


Figura 2-1. Ciclo de Vida JSF

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.5.5 Ventajas

- ✓ Está integrado dentro de páginas JSP, se encarga de recoger y generar valores de los elementos de la página.
- ✓ Permite introducir JavaScript, lo que acelera la respuesta de la interfaz en el cliente.
- ✓ Ofrece facilidad para que los desarrolladores y diseñadores puedan utilizarlo.
- ✓ Combina el rendimiento con un desarrollo rápido, permitiendo tener una mejor visualización de las interfaces.
- ✓ Permite desarrollar aplicaciones con frameworks de persistencia como por ejemplo JPA. (Llanos & Villa, 2011)

1.5.6 Desventajas

- × Su naturaleza como estándar hace que la evolución de JSF no sea tan rápida como pueda ser la de otros entornos como WebWork, Wicket, Spring, etc.

- × No existe suficiente documentación online actualizada.(Llanos & Villa, 2011)

1.6 Managed Beans

1.6.1 Definición

Es una clase que debe ser implementada con la interfaz serializable, cuyo objetivo es controlar el estado de las páginas web y se lo puede declarar de varias formas:

- Utilizando anotaciones, antes del nombre de la clase: @ManagedBean
- En el archivo faces-config.xml:

```
<managed-bean>Nombre del Managed Bean</managed-bean>
```

Los Managed Beans son creados por el framework JSF, además tienen asignado un identificador con el que es referenciado en las vistas. Para especificar el acceso desde las vistas JSF a las propiedades o los métodos a ejecutar de un Managed Beans se emplea la siguiente notación:

```
#nombreManagedBean.nombreAtributo
```

JSF se encarga de administrarlos automáticamente de la siguiente manera:

- Crea instancias, para lo cual requiere de un constructor vacío.
- Controla el ciclo de vida determinando el ámbito o alcance como es el caso de request, session, application, view.
- Se encarga de llamar a los métodos SET y GET.

1.6.2 Alcance

application	• Los objetos se encuentran disponibles para todas las peticiones de cualquier usuario en las vistas de la aplicación.
session	• Los objetos se encuentran disponibles para todas las peticiones que formen parte de la misma sesión de un cliente.
view	• Los objetos se encuentran disponibles para todas las peticiones que se realicen sobre la misma vista.
request	• Los objetos se encuentran disponibles desde que recibe una petición hasta que la respuesta se la envía al cliente.

Figura 3-1. Alcance MB

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.7 Primefaces

Fue desarrollada y mantenida por PrimeFace Technology que es una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. El proyecto es liderado por *Çağatay Çivici*, un miembro del "JSF Expert Group".

1.7.1 Definición

Es un componente para Java Server Faces de código abierto que cuenta con un conjunto de componentes ricos que facilitan la creación de las aplicaciones web. (Jaf, 2013)

1.7.2 Características

Es una potente biblioteca de componentes de interfaz de usuario muy fácil de utilizar que proporciona un conjunto de características:

- Proporciona soporte nativo de Ajax.
- Posee un conjunto de componentes ágiles como son el editor HTML, paneles, gráficas, entre otros.
- Ofrece un Kit para crear aplicaciones web móviles.
- Permite el uso de Javascript no intrusivo es decir no aparece en línea dentro de los elementos, sino dentro de un bloque `<script> </script>`
- Se caracteriza por ser open source, activo y bastante estable entre versiones. (Jaf, 2013)

1.7.3 Componentes

Ajax Core

Tabla 1-1: Etiquetas/componentes de Ajax Core

Etiquetas /Componentes	Descripción
Process	Permite la ejecución de determinados componentes sin tener en cuenta otros durante una petición.
Selector	Permite el uso de jQuery API de selección al hacer referencia a los componentes.
Poll	Permite realizar llamadas Ajax periódicamente.
Status	Es un indicador global para proporcionar información acerca de la petición Ajax en curso.
Counter	Es una variable de tipo entero que se incrementa cada vez que se hace clic en el botón y el valor actual se actualiza con el Ajax.
Validation	Las validaciones se ejecutan en el servidor y se actualiza el resultado en las páginas.
Search	Es el marco de búsqueda de expresiones, amplía dicha búsqueda por medio de palabras claves, además permite combinar las expresiones.
Fragment	Este componente se utiliza para definir parcial y automáticamente secciones del proceso y actualizar cada petición Ajax.

Fuente: (Mojarra, 2015)

Input

Tabla 2-1: Etiquetas/componentes de Input

Etiquetas /Componentes	Descripción
Autocomplete	Muestra sugerencias o varias opciones como contenido personalizable, selección múltiple, efectos y eventos.
BooleanButton	Se utiliza para seleccionar un valor booleano mediante un botón.
Calendar	Es un componente de entrada para recoger una fecha y una hora.
Editor	Es un componente de entrada con muchas características de edición de texto.
InputText	Es un elemento de entrada de tipo texto.
InputMask	Es un componente de entrada para ser formateado de manera determinada.
InputTextarea	Se extiende con AutoComplete AUTORESIZE, que puede mostrar el número de caracteres restantes
KeyFilter	Se puede utilizar para filtrar la entrada de teclado en los componentes de entrada especificados.
ListBox	Se utiliza para elegir un único elemento de una lista.
ManyButton	Se utiliza para seleccionar varios elementos de una lista usando los botones.
ManyCheckbox	Se utiliza para elegir varios artículos de una colección.
OneButton	Se utiliza para elegir un único elemento de una lista usando botones.
OneMenu	Se utiliza para elegir un único elemento de una lista.
OneRadio	Se utiliza para elegir un único elemento de una colección.
Password	Es una versión ampliada del componente inputSecret con el tema de integración e indicador de intensidad.
Slider	Se utiliza para proporcionar la entrada de varias maneras.

Fuente: (Mojarra, 2015)

Button

Tabla 3-1: Etiquetas/componentes de Button

Etiquetas /Componentes	Descripción
Button	Es una extensión de la etiqueta h:button, recibe peticiones dirigidas a las direcciones URL.
CommandButton	Representa el botón utilizado para enviar los datos introducidos en el campo de texto. El atributo action especifica una salida que facilita el mecanismo de navegación, es decir la decisión de que página abrir después.
CommandLink	Es una extensión de h:commandLink con Ajax, con procesamiento parcial y capacidad de desglose.
Link	Es un componente de navegación integrado con el modelo de navegación JSF.

Fuente: (Mojarra, 2015)

Data

Tabla 4-1: Etiquetas/componentes de Data

Etiquetas /Componentes	Descripción
Carousel	Permite mostrar un conjunto de datos o a su vez el contenido general.

DataGrid	Muestra los datos en una cuadrícula diseñada. Las características que integra es Ajax Pagination así como la interfaz de usuario puede ser totalmente personalizada.
DataList	Muestra un conjunto de datos en una lista con varios diseños y tipos de visualización. Posee características tales como: UnOrdered DataList, Ordered DataList, Ajax Pagination
DataExporter	Permite exportar tablas de datos a varios formatos como: Excel, pdf, xml, entre otros.
DataTable	Permite la paginación con Ajax, ordenar y filtrar varias soluciones de selección de encabezados, filas, desplazamiento en celdas.
MindMap	Es una herramienta interactiva que ofrece mapas mentales así como también animaciones.
OrderList	Se utiliza para clasificar una colección a través de arrastrar y reordenar.
PickList	Es un componente de entrada, se basa en el reordenamiento, efectos de transición, diseño sensible.

Fuente: (Mojarra, 2015)

Panel

Tabla 5-1: Etiquetas/componentes de Panel

Etiquetas /Componentes	Descripción
Accordion	Este componente contiene los paneles apilados verticalmente.
Dashboard	Es un componente de diseño que ofrece contenido basado en el reordenamiento similar a los entornos del portal.
fieldset	Es un componente del panel con una leyenda y un contenido.
Layout	Proporciona un diseño que puede ser aplicado a una página completa o para algunas partes de la misma, permite expandir, contraer, redimensionar y cerrar.
OutputPanel	Es un elemento contenedor con varios casos de uso como de actualización automática y carga diferida.
Panel	Es un componente de agrupación compuesto por: encabezado, contenido y pie de página.
PanelGrid	Es una extensión del h:panelGrid con el tema de agrupación, integración y características de respuesta.
TabView	Es utilizado para panel de pestañas, carga de contenido dinámico con Ajax, orientaciones diferentes, creación y eliminación de fichas mediante programación.

Fuente: (Mojarra, 2015)

Overlay

Tabla 6-1: Etiquetas/componentes de Overlay

Etiquetas /Componentes	Descripción
ConfirmDialog	Está integrado con el comportamiento de confirmación, se utiliza para sustituir la utilidad de confirmación de JavaScript
Dialog	Es un componente contenedor que puede incorporar otros elementos en la página. Proporciona varias opciones para cambiar el tamaño, altura y posición

LightBox	Es un componente que permite mostrar imágenes, contenido en líneas e iframes.
OverlayPanel	Es un contenedor genérico que puede sobreponer otros componentes en la página, puede presentarse en un panel básico, dinámico o a modo de imagen.
Tooltip	Muestra mensajes de ayuda.

Fuente: (Mojarra, 2015)

Menu

Tabla 7-1: Etiquetas/componentes de Menu

Etiquetas /Componentes	Descripción
ContextMenu	Es una pantalla de menú de superposición que aparece principalmente utilizando el botón derecho
MegaMenu	Muestra los submenús de elementos.
Menu	Este componente es personalizable, de comando y navegación que soporta posicionamiento dinámico y estático.
MenuBar	Proporciona barras de menú de aplicaciones de escritorio para JSF.
MenuButton	Agrupar varios comandos en un menú emergente.
PanelMenu	Es un componente híbrido de árbol-acordeón
SlideMenu	Muestra submenús anidados mediante la animación de diapositivas.
TabMenu	Componente menú que muestra ítems en pestañas
TieredMenu	Muestra submenús anidados.

Fuente: (Mojarra, 2015)

Charts

Tabla 8-1: Etiquetas/componentes de Charts

Etiquetas /Componentes	Descripción
Area	Es una versión personalizada de un LineChart.
Animate	En los gráficos se pueden realizar animaciones.
Bar	Este componente permite realizar diagramas de barras.
Export	Los gráficos se pueden exportar como imágenes estáticas en el lado del cliente.
Static	Es capaz de presentar imágenes que se crean mediante programación en tiempo de ejecución o las imágenes almacenadas en una base de datos.

Fuente: (Mojarra, 2015)

Message

Tabla 9-1: Etiquetas/componentes de Message

Etiquetas /Componentes	Descripción
growl, messages	Estos componentes se utilizan para mostrar mensajes.

Fuente: (Mojarra, 2015)

Multimedia

Tabla 10-1: Etiquetas/componentes de Multimedia

Etiquetas /Componentes	Descripción
Barcode	Es un componente que genera varios tipos de formatos png o svg
Compare	Proporciona una interfaz gráfica que permite comparar imágenes similares.
Cropper	Se utiliza para extraer cierta parte de una imagen para crear una nueva imagen.
Galleria	Es un componente que permite mostrar la galería de imágenes.
Media	Permite incrustar contenido multimedia como video y audio en las páginas jsf.
PhotoCam	Es un componente de entrada para tomar fotos con la cámara web y enviarlas al modelo back-end
Switch	Es un componente de la galería de imágenes con más de 25 efectos.

Fuente: (Mojarra, 2015)

Files

Tabla 11-1: Etiquetas/componentes de Files

Etiquetas /Componentes	Descripción
Upload	Se utiliza para la carga de archivos.
Download	Se utiliza para transmitir contenido binario como los archivos almacenados en la base de datos al cliente.

Fuente: (Mojarra, 2015)

DragDrop

Tabla 12-1: Etiquetas/componentes de DragDrop

Etiquetas /Componentes	Descripción
Draggable	Permite el movimiento de los componentes de forma horizontal, vertical, área definida, con efectos y otras características.
DataGrid	Permite la integración con el componente grid.
DataTable	Permite la integración con el componente tabla.

Fuente: (Mojarra, 2015)

Misc

Tabla 13-1: Etiquetas/componentes de Misc

Etiquetas /Componentes	Descripción
Captcha	Este componente permite crear un test controlado.
Collector	Es un componente para manejar colecciones sin código java.
Resizable	Permite cambiar el tamaño de cualquier componente.
Separator	Permite mostrar una línea horizontal como un componente individual, usado también como separador de menús y barra de herramientas.

Fuente: (Mojarra, 2015)

1.7.4 Ventajas

- ✓ Los componentes son amigables para el usuario, además proporciona una gran variedad de interfaces de usuario con diseños innovadores.
- ✓ Soporte Ajax transparente al desarrollador.
- ✓ Ofrece más de 100 componentes Open Source, siendo algunos de alta calidad y complejidad.
- ✓ Se puede elegir el tema de acuerdo a la preferencia puesto que está integrado con ThemeRoller Framework CSS.
- ✓ Puede ser utilizado con Richfaces y compatible con JSF 2.0
- ✓ Se actualiza muy rápidamente a comparación de otros frameworks. (Jaf, 2013)

1.7.5 Desventajas

- × Aplicaciones desarrolladas en una versión determinada podrían provocar un gran esfuerzo en poderla adecuar en otra versión.
- × Para utilizar Ajax tenemos que indicarlo explícitamente, por medio de atributos específicos de cada componente. (Jaf, 2013)

1.8 JDBC

JDBC puede acceder a cualquier tipo de datos tabulares, especialmente los datos almacenados en una base de datos relacional. Funciona con Java en una variedad de plataformas, además proporciona una base común para la construcción de herramientas y utilidades de alto nivel.

Para usar JDBC es necesario disponer de un driver apropiado, dependiendo de varios factores, este puede estar escrito en Java puro o puede ser una mezcla de Java y métodos nativos JNI (Java Native Interface). (Tutorialspoint, 2016)

1.8.1 Definición

Consiste en un conjunto de clases e interfaces escritas en el lenguaje de programación Java. JDBC suministra el API para los desarrolladores esto hace posible escribir aplicaciones de base de datos.

La combinación de Java y JDBC permite al desarrollador escribir una sola vez y ejecutarlo en cualquier entorno.

1.8.2 ¿Qué hace JDBC?

- Establece una conexión con una base de datos.
- Envía sentencias SQL.
- Procesa los resultados. (Tutorialspoint, 2016)

1.8.3 Características

- Es un API de bajo nivel, es decir permite ejecutar comandos SQL directamente y procesar los resultados obtenidos.
- Permite que cualquier comando SQL pueda ser pasado al driver directamente, por lo que una aplicación desarrollada en Java pueda hacer uso de toda la funcionalidad que provea el motor de base de datos.
- Ha incluido una gran cantidad de métodos, en vez de hacer métodos complejos con gran cantidad de parámetros.(Pincay, 2013)

1.9 PostgreSQL

Según investigaciones previas PostgreSQL ha sido definido como uno de los RDBMS más utilizados cuyo objetivo fundamental es desarrollar aplicaciones en las que predomine la estabilidad, la integridad y la seguridad del sistema.

1.9.1 Definición

Es una de las bases de datos más avanzadas y potentes del mercado de código abierto, que utiliza la arquitectura cliente/servidor y tiene la capacidad de competir con muchos gestores comerciales.(Mancero, 2013)

1.9.2 Arquitectura

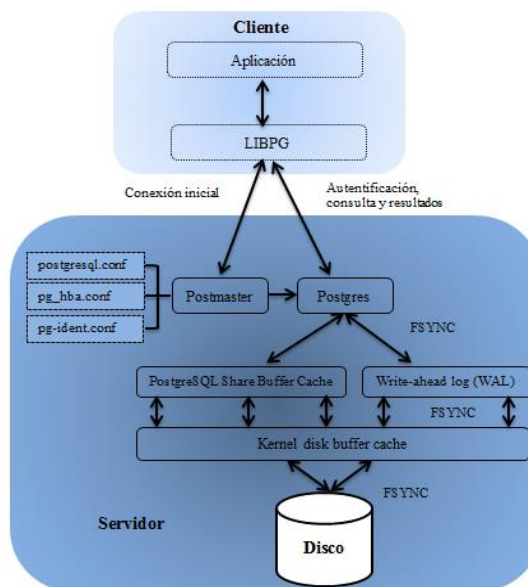


Figura 4-1. Arquitectura de PostgreSQL

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Se detalla a continuación cada uno de los componentes que conforman la Arquitectura de PostgreSQL que se visualiza en la Figura 4-1:

Aplicación Cliente.- utiliza como administrador de base de datos PostgreSQL, además la conexión se realiza vía TCP/IP o por medio de sockets locales.

PostMaster.- es el responsable de crear los procesos hijos que se encargarán de autenticar las peticiones, gestionar las consultas y enviar los resultados, así como también es el encargado de escuchar las conexiones entrantes de los clientes por medio de un puerto o socket.

Ficheros de Configuración.- los ficheros utilizados se detallan a continuación:

- postgresql.conf: los cambios que se realicen en el fichero afectarán a todas las bases de datos que tengamos definidas en el clúster PostgreSQL.
- pg_hba.conf: este fichero se utiliza para definir cómo, dónde y desde que sitio puede utilizar el clúster PostgreSQL.
- pg_ident.conf: este fichero se utiliza para definir la información necesaria en el caso que utilicemos un acceso de tipo ident en pg_hba.conf

Procesos hijos postgres.- estos procesos se encargarán de autenticar a los clientes, gestionar las consultas y enviar los resultados.

PostgreSQL share buffer cache.- es la memoria compartida utilizada por PostgreSQL para almacenar datos en caché.

Write-Ahead Log (WAL).- es el componente del sistema encargado de asegurar la integridad de los datos.

Kernel disk buffer cache: es la caché de disco del sistema operativo.

Disco: es el disco físico donde se almacenan los datos y toda la información necesaria para que PostgreSQL funcione. (Mancero, 2013)

1.9.3 Características

- Es una base de datos 100% orientada ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).
- Proporciona Integridad referencial.
- Permite el almacenamiento de objetos binarios grandes como gráficos, videos y sonido.
- Utiliza llaves primarias o conocidas como primary keys y las llaves foráneas conocidas como foreign keys.
- Permite la creación de check, unique y not null constraints.
- Permite la creación de índices compuestos, únicos, parciales y funcionales en cualquier método de almacenamiento disponible.
- Permite la utilización y creación de consultas recursivas, subconsultas o sub-selects.
- Permite la creación de vistas, triggers, procedimientos almacenados. (Mancero, 2013)

1.9.4 Límites

Existen varios límites a tener en cuenta antes de seleccionar dicho gestor de base de datos, los mismos que se visualizan en la Tabla 14-1:

Tabla 14-1: Límites PostgreSQL

Límite	Valor
Máximo tamaño de base de datos	(Ilimitado) pero depende del Sistema de Almacenamiento
Máximo tamaño de base de tabla	32 TB
Máximo tamaño de base de fila	1.6 TB
Máximo tamaño de base de campos	1 GB
Máximo tamaño de base de filas por tabla	Ilimitado
Máximo tamaño de base de columnas por tabla	250 – 1600
Máximo tamaño de base de índices por tabla	Ilimitado

Fuente: (Mancero, 2013)

1.9.5 Ventajas

- ✓ Posee gran capacidad de almacenamiento.
- ✓ Posee licencia gratuita.
- ✓ Ofrece ahorros considerables en costos de operación.
- ✓ Ofrece estabilidad, escalabilidad y rendimiento.
- ✓ Opera sobre distintas plataformas.
- ✓ Proporciona herramientas gráficas de diseño y administración.
- ✓ Es extensible y la instalación es ilimitada. (Grupo, 2013)

1.9.6 Desventajas

- × Utiliza y consume muchos más recursos que MySQL.
- × Es un poco lento al realizar las inserciones y actualizaciones.
- × La sintaxis de algunos de las sentencias o comandos no es nada intuitiva.
- × Proporciona una reducida cantidad de tipos de datos. (Grupo, 2013)

1.10 JPA

Es la API de persistencia desarrollada para la plataforma Java EE, se encarga de realizar la conversión entre los objetos y tablas de una base de datos, dicha conversión se la denomina Mapeo Relacional de Objetos (ORM) y puede configurarse a través de metadatos mediante anotaciones o XML. (Tutorialspoint, 2016)

1.10.1 Definición

Es el conjunto de clases y métodos para almacenar persistentemente la enorme cantidad de datos en una base de datos. (Tutorialspoint, 2016)

Está construida alrededor de tres áreas principales:

- API de persistencia Java.
- Metadatos de transformación objeto/relacional (Anotaciones).
- Lenguaje de Consulta JPQL

1.10.2 Arquitectura

La siguiente figura muestra las clases principales y las interfaces de JPA:

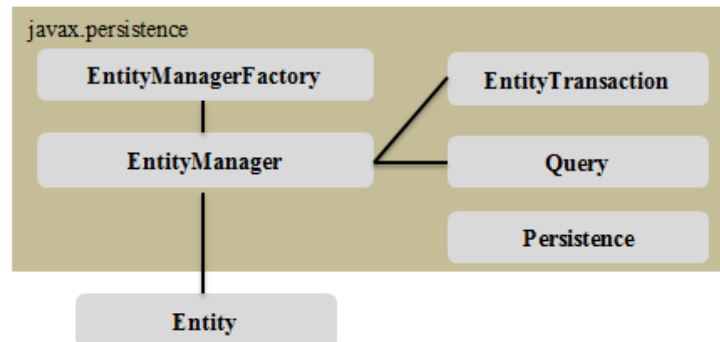


Figura 5-1. Arquitectura Externa JPA

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.10.3 Elementos

Para facilitar la persistencia JPA se basa en los siguientes elementos:

- Entidades
- Contexto de persistencia
- Unidad de persistencia
- Administrador de Entidades (Chavarria, 2013)

1.10.3.1 Entidades

Es un objeto persistente generalmente está relacionado con una tabla en una base de datos relacional y cada instancia del Entity se asocia con una fila en dicha tabla.

Entidad = POJO (Plain Old Java Object) + anotaciones

La estructura de una entidad se detalla de la siguiente manera:

```
@Entity
public class Persona {
    // instrucciones
}
```

Pojo es una instancia de una clase que no extiende ni implementa nada en especial. Los programadores utilizan para enfatizar el uso de clases simples y que no dependen de un framework especial. (Chavarria, 2013)

1.10.3.1.1 Ciclo de Vida

La entidad puede encontrarse en cualquiera de estos cuatro estados:

1. Nueva: cuando la entidad se acaba de instanciar el operador new y no está asociado a ningún contexto de persistencia.
2. Gestionada: tiene una identidad persistente y está asociado a un contexto de persistencia.
3. Separada: tiene una identidad persistente pero no está asociado a un contexto de persistencia
4. Eliminada: tiene una identidad persistente y está asociada a un contexto de persistencia, pero está marcada para ser eliminada de la base de datos. (Chavarria, 2013)

1.10.3.1.2 Anotaciones

A continuación en la Tabla 15-1 se detallan las principales anotaciones:

Tabla 15-1: Anotaciones

Anotación	Descripción
@Entity	Representa una entidad.
@Table	Especifica la tabla principal relacionada con la entidad.
@SecondaryTable	Especifica una tabla secundaria relacionada con la entidad, tiene los mismos atributos que @Table.
@UniqueConstraints	Especifica que una única restricción se incluye tanto para la tabla principal y secundaria.
@Column	Especifica una columna de la tabla a mapear con un campo de la entidad.
@JoinColumn	Especifica un campo de la tabla que es foreign key de otra tabla.
@Id	Indica la clave primaria de la tabla.
@GeneratedValue	Se asocia con la clave primaria.
@SequenceGenerator	Define un generador de claves primarias utilizado junto con la anotación @GeneratedValue.
@IdClass:	Se aplica en la clase entidad para especificar una composición de la clave primaria mapeada a varios campos o propiedades de la entidad.
@Basic	Mapeo por defecto para los tipos de datos básicos.
@JoinTable:	Se utiliza en el mapeo de una relación ManyToMany o en una relación unidireccional OneToMany.
@NamedQuery:	Especifica el nombre del objeto query utilizado junto al EntityManager.
@NamedQueries:	Especifica varias queries.
@NamedNativeQuery	Especifica el nombre de una query SQL normal
@NamedNativeQueries:	Especifica varias queries SQL

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.10.3.1.3 Relación entre entidades

Uno a uno (@OneToOne).- en esta relación cada instancia de un entity se relaciona con una única instancia de otro entity.

Uno a muchos (@OneToMany).- una instancia de un entity puede estar relacionada con múltiples instancias de otros entities.

Muchos a uno (@ManyToOne).- múltiples instancias de un entity pueden estar relacionadas con una única instancia de otro entity. Es la opuesta a la relación uno a muchos.

Muchos a muchos (@ManyToMany).- varias instancias de un entity pueden estar relacionadas con múltiples instancias de otros entities. (Escobar, 2011)

1.10.3.2 Contexto de Persistencia

Es un conjunto de instancias que existen en un almacén de datos en las que para cualquier entidad hay únicamente una instancia. En el contexto de persistencia las instancias de las entidades y los ciclos de vida son administrados y se asocian con una unidad de persistencia.

(Escobar, 2011)

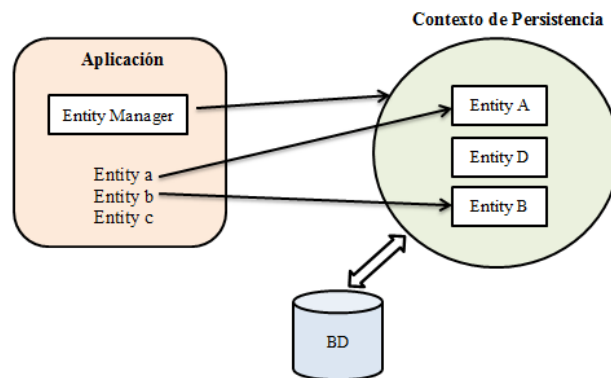


Figura 6-1. Contexto de Persistencia

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

1.10.3.3 Unidad de Persistencia

Representa un conjunto de entidades que pueden ser mapeadas a una base de datos, que trabajarán de forma conjunta y compartirán una misma configuración así como la información necesaria para que la aplicación JPA pueda acceder a dicha base de datos.

Las unidades de persistencia se definen empleando un archivo XML denominado persistence.xml y deben asociarse a los objetos EntityManager, las mismas que deben ser identificadas con un nombre único en el ámbito de la unidad de persistencia. (Escobar, 2011)

Persistence.xml.- en este archivo de configuración se define los contextos de persistencia de la aplicación, el mismo que se encuentra dentro del directorio META-INF.

```

<?xml version="1.0" encoding="UTF-8"?>
<persistence version="2.1" xmlns="http://xmlns.jcp.org/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd">
  <persistence-unit name="SGAF_GENIOS_TRABAJANDOPU" transaction-type="RESOURCE_LOCAL">
    <provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>
    <class>ec.edu.genios.entidades.Provincia</class>
    <class>ec.edu.genios.entidades.Contabilidaddiariageneral</class>
    <class>ec.edu.genios.entidades.Sucursal</class>
    <class>ec.edu.genios.entidades.Calculopago</class>
    <class>ec.edu.genios.entidades.Capacitacion</class>
    <class>ec.edu.genios.entidades.Periodo</class>
    <class>ec.edu.genios.entidades.Institucion</class>
    <class>ec.edu.genios.entidades.Contabilidadmensual</class>
    <class>ec.edu.genios.entidades.Inscripcion</class>
    <class>ec.edu.genios.entidades.Calculopagogenios</class>
    <class>ec.edu.genios.entidades.Ciudad</class>
    <class>ec.edu.genios.entidades.Docente</class>
    <class>ec.edu.genios.entidades.Cobro</class>
    <class>ec.edu.genios.entidades.Tipohorario</class>
    <class>ec.edu.genios.entidades.Tipocobro</class>
    <class>ec.edu.genios.entidades.Usuario</class>
    <class>ec.edu.genios.entidades.Horario</class>
    <properties>
      <property name="javax.persistence.jdbc.url" value="jdbc:postgresql://localhost:5433/BaseSGAF_GENIOS"/>
      <property name="javax.persistence.jdbc.user" value="postgres"/>
      <property name="javax.persistence.jdbc.driver" value="org.postgresql.Driver"/>
      <property name="javax.persistence.jdbc.password" value="postgres"/>
      <property name="eclipselink.allow-zero-id" value="true"/>
    </properties>
  </persistence-unit>
</persistence>

```

Figura 7-1. Archivo persistence.xml

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Los elementos más importantes del archivo persistence.xml son los siguientes:

persistence-unit name: especifica el nombre para el contexto o la persistencia.

transaction-type: el valor para este elemento es JTA o RESOURCE-LOCAL. El tipo de transacción por defecto es RESOURCE-LOCAL para aplicaciones Java Standar Edition.

provider: se encarga de especificar el nombre del proveedor de persistencia. Por ejemplo: Toplink Essentials, Hibernate, EclipseLink.

class: muestra el listado de los nombres de las entidades que son parte de la unidad de persistencia.

properties: se especifica el tipo de base de datos a utilizar en entornos JSE, las propiedades de conexión a la base de datos incluyen:

- Nombre de usuario
- Contraseña
- Cadena de conexión (URL)
- Nombre de la clase del driver. (Escobar, 2011)

Entity Manager

El administrador de Entidades es una interfaz que se apoya en el API de persistencia y se encarga de del mapeo entre un tabla relacional y su objeto Java, además de proporcionar métodos para manejar la persistencia de una entidad.

Entre los principales métodos se detallan los siguientes:

Tabla 16-1: Métodos Entity Manager

Métodos	Descripción
<code>persist (Object entity)</code>	Se encarga de almacenar el objeto entity en la base de datos.
<code>merge (T entity)</code>	Se encarga de actualizar la entidad devolviendo la lista resultante.
<code>remove (Object entity)</code>	Se encarga de eliminar la entidad.
<code>find(Class entity, Object primaryKey)</code>	Se encarga de buscar la entidad a través de su clave primaria.
<code>refresh (Object entity)</code>	Refresca el estado de la entidad con su contenido en la base de datos.
<code>createQuery(String query)</code>	Crea un query utilizando el lenguaje JPQL
<code>createNativeQuery()</code>	Crea un query utilizando el lenguaje SQL
<code>isOpen()</code>	Comprueba si el EntityManager está abierto
<code>close()</code>	Cierra el EntityManager

Fuente: (Caules, 2013)

Consultas Nativas SQL

A los comandos SQL en JPA se los conoce con el nombre de consultas nativas en vez de utilizar JPQL, es decir al utilizar comandos SQL en la base de datos se puede aprovechar algunas características del motor de base de datos.

La desventaja que se presenta es que un comando nativo utilizado en una base de datos podría no funcionar correctamente en otra. Para ello se utiliza el método `createNativeQuery`. (Gómez, 2013)

Ejemplo:

```
String sql = "SELECT * FROM estudiante"
Query query = em.createNativeQuery(sql);
```

1.10.3.4 Ventajas

- ✓ Utiliza una única clase para declarar la persistencia mediante la ayuda de las anotaciones.
- ✓ Las clases a persistir son simples POJOS.
- ✓ Se puede usar en entornos Java Standar Edition, así como en Java Enterprise Edition.
- ✓ Permite utilizar diferentes proveedores de persistencia sin afectar el código del Entity.
- ✓ Facilidad de aprendizaje
- ✓ Evita construir sentencias SQL, ya que mediante una configuración XML y anotaciones sobre el código fuente se puede mapear directamente objetos de las tablas de base de datos.

1.10.3.5 Desventajas

- × El uso de anotaciones dificultan un poco el mantenimiento.
- × Su uso es limitado para la comunidad de Java, pero el API de persistencia puede aparecer en las otras plataformas en un futuro. (Escobar, 2011)

1.11 Proveedor de Persistencia

Los proveedores de persistencia son implementaciones de la especificación JPA y se pueden desplegar en el servidor de aplicaciones compatible con Java EE.

1.11.1 EclipseLink

Es una solución completa de persistencia de código abierto. Fue iniciado por una donación de las suites de código fuente y las pruebas completas de productos de Oracle TopLink. Este proyecto trae la experiencia de más de 12 años de uso comercial y el desarrollo de funciones Java.

Proporciona una solución completa, ya que no proporciona una, sino un conjunto de servicios de persistencia que permite a los desarrolladores crear eficientemente aplicaciones que acceden a datos en una variedad de fuentes de datos y formatos.

Permite una amplia integración y minimiza el acoplamiento, ofrece una mayor flexibilidad cuando se trata de esquemas relacionales complejos. EclipseLink es compatible con una serie de normas de persistencia incluida la API de persistencia de Java (JPA), Java API para XML, Arquitectura Java Connector, Service Data Objects. (EcuRed, 2016)

1.12 Ireport

Nació como una herramienta de desarrollo, sin embargo puede utilizarse como una herramienta de oficina para adquirir u obtener datos almacenados en una base de datos, sin la necesidad de utilizar otra aplicación.

1.12.1 Definición

Es una herramienta visual de tipo open source usada para crear informes por medio de JasperReport, está implementado completamente en Java, permitiendo que los usuarios puedan corregir automáticamente los errores que se presentan en los informes complejos como las cartas, imágenes o subinformes. (EcuRed, 2016)

1.12.2 Características

- Está 100% escrito en Java, además de ser libre y gratuito.
- Maneja el 98% de las etiquetas de JasperReport.
- Permite el diseño de rectángulos, líneas, elipses, textfields, cartas, subreportes, entre otros mediante sus propias herramientas.
- Soporta JDBC y posee facilidad de instalación.
- Incluye Wizard's para crear automáticamente informes. (EcuRed, 2016)

1.12.3 Funciones

Es una biblioteca que puede ser embebida en cualquier aplicación Java. Entre las principales funciones se detallan a continuación:

Scriptlets.- pueden ser invocados en cualquier momento para realizar el procesamiento adicional y pueden acompañar a la definición del informe.

Sub-informes.- es un elemento de un informe que muestra otro informe dentro del cuerpo del informe principal. (EcuRed, 2016)

Las principales librerías que se utilizan son las siguientes:

- jasperreports-1.0.1.jar
- commons-digester.jar
- commons-beanutils.jar
- commons-collections.jar
- commons-logging.jar
- itext-1.02b.jar

CAPÍTULO II

2 MARCOMETODOLÓGICO

2.1 Diseño de la Investigación

El diseño de la investigación “es el pegamento que mantiene el proyecto de investigación cohesionado. Un diseño es utilizado para estructurar la investigación, para mostrar cómo todas las partes principales del proyecto de investigación funcionan en conjunto con el objetivo de responder a las preguntas centrales de la investigación”. El diseño de la investigación es la “columna vertebral” del protocolo de investigación. (López, 2015,p1)

El diseño de la investigación se refiere a la forma, método, estructura que el investigador utilizará para alcanzar los objetivos de su estudio, ya que describe los pasos a seguir en la investigación y de esta forma determinar la calidad de la misma. (Zulay, 2011)

El presente trabajo de titulación es un proyecto técnico en donde se aplicarán los conocimientos adquiridos en el transcurso de la vida universitaria la cual se lo ha realizado con el propósito de dar solución al problema encontrado en la institución teniendo como objetivo el “Diseño e implementación de una Aplicación Web para la Gestión Académica y Financiera para el Centro de Capacitación Académica y Preuniversitaria “Genios Trabajando” mediante la utilización de los frameworks PrimeFace/JPA” para esto se ha realizado un estudio previo de los frameworks a emplearse, de igual forma se ha recolectado información necesaria mediante métodos y técnicas de recolección.

2.2 Métodos

Para el desarrollo de la aplicación web se utilizará los siguientes métodos basados en el análisis y síntesis que ayudarán a determinar y establecer las diferentes necesidades que se presentan en dicha institución.

Análisis

Consiste en la descomposición de un todo en sus elementos, es decir separar los procesos que se llevan a cabo en la institución para desarrollarlos de forma individual.

Síntesis

Consiste en la recopilación de toda la información obtenida mediante el análisis, identificando los principales procesos para el desarrollo del sistema para la Gestión Académica y Financiera de la institución.

2.3 Técnicas

Para la recopilación de información que sustente el desarrollo de la aplicación se utilizará las siguientes técnicas detalladas a continuación:

- Revisión de documentos que permitan determinar los procesos que se realizan en el Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”, los mismos que se detallan en el Anexo A.
- Revisión de Documentos Digitales acerca de Primefaces y JPA.
- Observación.- El instrumento que se utilizará son las fichas de observación que nos permitirá registrar datos de los tiempos y errores antes y después de la implementación del sistema y de esta forma determinar los beneficios obtenidos.
- Encuesta es realizada a los usuarios que utilizarán el sistema, con la finalidad de medir la usabilidad del mismo.

2.4 Metodología Scrum

Para el desarrollo de la aplicación web se utilizará la metodología SCRUM puesto que es una metodología fácil de implantar y muy popular por los resultados rápidos que consigue, además se aplica a un conjunto de mejores prácticas para trabajar en equipo, es útil para proyectos complejos donde los requisitos son cambiantes, es decir existe flexibilidad y adaptación de acuerdo a las necesidades del cliente, garantiza la entrega de un producto una vez finalizada cada iteración y así el cliente comprueba si se van cumpliendo sus expectativas. (Mosto, 2011)

2.4.1 Ciclo de Vida

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración de entre 2 y 4 semanas obteniendo como resultado una versión del software con nuevas funcionalidades para ser usadas.



Figura 1-2. Ciclo de Vida SCRUM

Fuente: (Silva, 2012)

2.4.2 Artefactos

Los artefactos de la metodología Scrum son útiles y están diseñados para maximizar la transparencia para la inspección y adaptación de la información, se detallan a continuación:

Product Backlog.- es el conjunto de requisitos, características, funcionalidades denominadas historias de usuario, los mismos que están escritos en un lenguaje no técnico, los ítems de mayor prioridad deben ser completados por el equipo lo antes posible. Los requisitos y prioridades se revisan y ajustan durante el desarrollo del proyecto por el Product Owner, para maximizar el éxito para el negocio de los esfuerzos del equipo.

Sprint Backlog.- es la lista de las tareas necesarias en las que el equipo trabaja para convertir las historias de usuario del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo del sprint, este plan hace visible todo el trabajo que el equipo de desarrollo ha identificado como necesario para cumplir con el objetivo del Sprint.

Burndown chart.- es el gráfico que muestra la cantidad de trabajo restante del Sprint, es decir controla el progreso del mismo, permite la re-estimación y la adición de nuevas tareas. Este artefacto es muy importante para los Stakeholders. (Mosto, 2011)

2.4.3 Roles



Figura 2-2. Roles SCRUM

Fuente: (Cprime, 2012)

Scrum master.- es la persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner.

Product Owner.- es el representante de los accionistas y clientes que utilizan el software. Se centra en la parte del negocio y es el responsable del proyecto.

Team members.- es el grupo de profesionales integrado por programadores, diseñadores, arquitectos, testers y demás con los conocimientos técnicos requeridos para que desarrollen el proyecto de manera conjunta con la realización de las historias de usuario.(Bahit, 2011)

Stakeholder.- puede ser cualquier persona que tenga un interés ya sea directo o indirecto en el trabajo del equipo.

2.5 Metodología Scrum para el desarrollo del Sistema

A continuación se detallará la metodología SCRUM para desarrollar el Sistema de Gestión Académica y Financiera “SGAF” para el Centro de Capacitación Académica y PreUniversitaria “GENIOS TRABAJANDO”

2.5.1 Planificación

En la fase de planificación del sistema se realiza la selección de requisitos, prioridades de las historias de usuario así como también el listado de las tareas realizadas.

Tabla 1-2: Tareas Realizadas

Institución	Tareas realizadas	Responsable
Centro de Capacitación Académica y Preuniversitario “GENIOS TRABAJANDO”	Entrevista y Acuerdos con la directiva.	Mercy Sani
	Recolección de Información	
	Diseño del proyecto.	
	Arquitectura de sistema	
	Tecnologías para el desarrollo	
	Estándar de Codificación	
	Diseño de la base de datos	Patricia Allauca
	Definición de Interfaz de Usuario	
	Desarrollo del Sistema	
	Presentación y refinamiento de las diferentes interfaces.	
	Detección y corrección de errores.	
	Presentación del sistema a la directiva.	
	Instalación del Sistema	
	Capacitación a los Usuarios.	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.1.1 Personas y Roles

Se detalla las personas y los roles respectivamente en la Tabla 2-2 que se muestra a continuación:

Tabla 2-2: Recurso Humano - Roles

RECURSO HUMANO – ROLES	
Scrum Master	Dr. Julio Santillán
Desarrolladores	Patricia Allauca – Mercy Sani
Product Owner	Ing. Diego Mayorga

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.1.2 Reuniones

Sprint Planning

En esta reunión el equipo determina la cantidad de historias que puede comprometerse a finalizar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.

Esta reunión se la realizó el 06/11/2015 en la institución ubicada en la calle Eugenio Espejo 21-55 y 10 de Agosto, en la que participaron las siguientes personas:

Directivo de la institución: Ingeniero Diego Mayorga

Equipo de trabajo: Mercy Sani y Patricia Allauca

En la cual se determinó las actividades iniciales a desarrollarse siendo un total de 5 requerimientos que corresponden al Sprint 1.

Daily Sprint

Esta reunión se la realiza diariamente con una duración de 15 minutos para exponer los problemas encontrados o el progreso alcanzado en la que el equipo se sincroniza para trabajar de forma coordinada, es decir que cada miembro del equipo expone 3 cosas:

1. Que es lo que hizo el día anterior.
2. Lo que realizará hoy.
3. Si existe algún inconveniente que no le permita completar el trabajo.

Esta reunión se la realizó el 05/02/2016 en la calle Mariana de Jesús y Cordovez, en la que participaron los miembros del equipo de desarrollo:

Patricia Allauca realizó la función que permite realizar el cálculo pago de los docentes manifestó que el valor total obtenido de las proyecciones en la función no coincidían con los valores de la tabla proyeccionpago de la base de datos. Lo que se procede a realizar es dar mantenimiento a la función es decir encontrar y solucionar el inconveniente antes mencionado.

Mercy Sani realizó los métodos necesarios para poder realizar la devolución de los cobros realizados a los estudiantes, encontrando como inconveniente el error al cargar los cobros de los estudiantes mostrándose datos anteriores no necesarios. Se procede a dar mantenimiento al método mencionado para que se muestren solo datos actuales.

Cabe recalcar que los cambios realizados se entregarán en la siguiente reunión.

2.5.1.3 Product Backlog

Los requerimientos del sistema se conocen con el nombre de Product Backlog los mismos que se detallan a continuación en la Tabla 3-2:

Tabla 3-2: Product Backlog

ID	DESCRIPCIÓN
HU1	Como director deseo ingresar los datos de los estudiantes.
HU2	Como director deseo ingresar los datos de los docentes (Capacitación/Nivelación).
HU3	Como director deseo ingresar los datos de las instituciones educativas.
HU4	Como director deseo ingresar los períodos académicos (Capacitación/Nivelación).
HU5	Como director deseo ingresar los cursos (Capacitación/Nivelación).
HU6	Como director deseo ingresar sucursales.
HU7	Como director deseo ingresar los horarios (Capacitación/Nivelación).
HU8	Como director deseo ingresar las aulas.
HU9	Como director deseo ingresar las capacitaciones.
HU10	Como director deseo registrar nuevos usuarios para el acceso al sistema.
HU11	Como director deseo registrar las inscripciones de los estudiantes.
HU12	Como director deseo asignar docentes y aulas a los estudiantes inscritos
HU13	Como director deseo registrar e imprimir los cobros de los estudiantes
HU14	Como director deseo registrar los gastos
HU15	Como director deseo realizar la proyección de los cursos.
HU16	Como director deseo calcular el pago de los docentes
HU17	Como director deseo modificar los datos de los estudiantes.
HU18	Como director deseo modificar los datos de las aulas.
HU19	Como director deseo modificar los datos de los docentes.
HU20	Como director deseo modificar los datos de las instituciones educativas.
HU21	Como director deseo modificar los períodos académicos.
HU22	Como director deseo modificar los cursos.
HU23	Como director deseo modificar los datos de las sucursales
HU24	Como director deseo modificar capacitaciones.
HU25	Como director deseo modificar los datos de usuarios.
HU26	Como director deseo realizar el cambio de curso de los estudiantes.
HU27	Como director deseo realizar la devolución de las cuotas si el estudiante se ha retirado.
HU28	Como director deseo generar el reporte de los pagos efectuados por los estudiantes dado un curso.
HU29	Como director deseo generar el reporte de los estudiantes dado un curso.
HU30	Como director deseo listar las instituciones.
HU31	Como director deseo listar los cursos.
HU32	Como director deseo listar los horarios.
HU33	Como director deseo listar los docentes.
HU34	Como director deseo listar las capacitaciones.
HU35	Como director deseo listar los usuarios.
HU36	Como director deseo generar e imprimir el rol de pagos para el docente
HU37	Como director deseo registrar y emitir el reporte de la contabilidad diaria.
HU38	Como director deseo registrar y emitir el reporte de la contabilidad mensual
HU39	Como director deseo emitir el reporte de los pagos a los docentes dada una fecha
HU40	Como director deseo la autenticación de usuarios.

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.1.4 Plan de Entrega

En el Plan de Entregas que se visualiza en la Tabla 4-2 se define las historias de usuario correspondiente a los respectivos Sprint creados. A los Sprints detallados el tiempo que se los ha asignado está entre 4 a 5 semanas dependiendo de la complejidad o facilidad de los mismos. Mediante la utilización de la metodología SCRUM, obtenemos cinco Sprints como se muestra a continuación:

Tabla 4-2: Requerimientos del Sistema

Sprints	Id	Descripción	Fecha Inicio	Fecha Fin
SPRINT 1.- Actividades Iniciales	HT1	Arquitectura del Sistema	09/11/2015	04/12/2015
	HT2	Definición de las tecnologías de desarrollo		
	HT3	Definición del Estándar de Codificación		
	HT4	Diseño de la base de datos		
	HT5	Definición de la Interfaz de Usuario.		
SPRINT 2.- Académico – Ingresos	HU1	Registrar los datos de los estudiantes.	07/12/2015	01/01/2016
	HU2	Ingresar los datos de los docentes (Capacitación/Nivelación).		
	HU3	Ingresar los datos de las instituciones educativas.		
	HU4	Ingresar los períodos académicos (Capacitación/Nivelación).		
	HU5	Ingresar datos de los cursos (Capacitación/Nivelación).		
	HU6	Ingresar las sucursales.		
	HU7	Ingresar los horarios (Capacitación/Nivelación).		
	HU8	Ingresar las aulas.		
	HU9	Ingresar las capacitaciones.		
	HU10	Registrar nuevos usuarios al sistema.		
	HU11	Registrar las inscripciones de los estudiantes.		
	HU12	Asignar el docente y el aula a los estudiantes que ya están inscritos.		
	HU14	Registrar los gastos		
SPRINT 3.- Académico – Modificaciones y Listados	HU17	Modificar los datos de los estudiantes.	04/01/2016	29/01/2016
	HU18	Modificar los datos de las aulas.		
	HU19	Modificar los datos de los docentes.		
	HU20	Modificar los datos de las instituciones educativas.		
	HU21	Modificar los datos de los períodos académicos.		
	HU22	Modificar datos de los cursos.		
	HU23	Modificar los datos de las sucursales		

	HU24	Modificar datos de las capacitaciones.		
	HU25	Modificar los datos de usuarios.		
	HU26	Realizar el cambio de curso de los estudiantes.		
	HU30	Listar las instituciones.		
	HU31	Listar los datos de los cursos actuales.		
	HU32	Listar los horarios.		
	HU33	Listar los docentes.		
	HU34	Listar las capacitaciones.		
	HU35	Listar los usuarios.		
	HU40	Autenticación de usuarios para acceso al sistema.		
Sprint 4.- Financiero	HU13	Registrar e imprimir los cobros de los estudiantes	01/02/2016	26/02/2016
	HU15	Realizar la proyección de los cursos.		
	HU16	Calcular el pago de los docentes		
	HU27	Realizar la devolución de las cuotas si el estudiante se ha retirado.		
Sprint 5.- Reportes	HU28	Generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.	29/02/2016	31/03/2016
	HU29	Generar el reporte de los estudiantes dado un curso.		
	HU36	Generar e imprimir el rol de pagos para el docente		
	HU37	Registrar y emitir el reporte de la contabilidad diaria.		
	HU38	Registrar y emitir el reporte de la contabilidad mensual		
	HU39	Emitir el reporte de los pagos a los docentes dada una fecha		

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2 Desarrollo

En esta fase se procede a definir las actividades iniciales para el desarrollo de la aplicación web, las mismas que se detallan a continuación:

2.5.2.1 Sprint 1

Al iniciar el desarrollo del sistema se procede a realizar las diferentes historias técnicas requeridas para continuar con la implementación del sistema y se detallan a continuación:

2.5.2.1.1 Arquitectura del Sistema

HT1: Definición de la arquitectura del sistema

En el desarrollo del sistema se inició con la realización de la primera historia técnica que está relacionada con la definición de la arquitectura del sistema como se detalla en la Tabla 5-2.

Tabla 5-2: Arquitectura del Sistema

HISTORIA TÉCNICA	
Número: HT1	Sprint Asignado: 1
Tiempo Estimado:	4 horas
Responsable:	Patricia Allauca – Mercy Sani
Descripción: Como administrador del sistema solicito se defina la arquitectura del sistema.	
Pruebas de Aceptación: <ul style="list-style-type: none">- La arquitectura del sistema tiene que ser realizada por los miembros del equipo de trabajo, (adecuada realización del diagrama de despliegue).	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

La arquitectura que se empleó para el diseño del sistema web es: Cliente/Servidor

Cliente: es el que inicia un requerimiento de servicio. El requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de redes LAN o WAN. La ubicación de los datos o de las aplicaciones es totalmente transparente para el cliente. (EcuRed, 2015)

Servidor: es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente. Los servidores pueden estar conectados a los clientes a través de redes LANs o WANs, para proveer de múltiples servicios a los clientes y usuarios tales como impresión, acceso a bases de datos, fax, procesamiento de imágenes, etc. (EcuRed, 2015)

2.5.2.1.1.1 Diagrama de Despliegue

Permite modelar la arquitectura en tiempo de ejecución de un sistema, se lo utiliza para representar las relaciones físicas entre sus componentes hardware y software. Se lo puede visualizar en la Figura 3-2

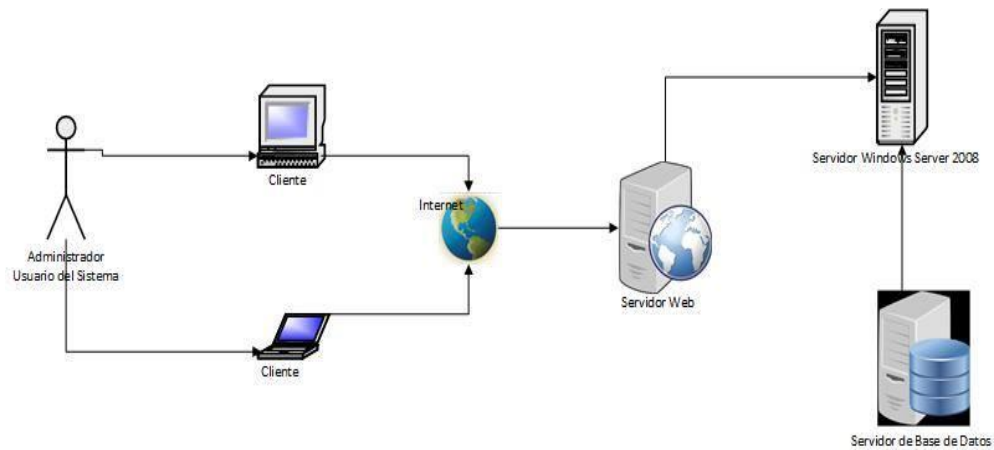


Figura 3-2. Arquitectura del Sistema

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.1.1.2 Diagrama de Componentes

Un diagrama de componentes hace parte de la vista física del sistema, la cual modela la estructura de implementación de la aplicación, organización en componentes, es decir se usa para mostrar las interfaces, para lo cual usa flechas con líneas discontinuas desde los componentes a las interfaces de otros componentes.

En la Figura 4-2 muestra el diagrama de componentes:

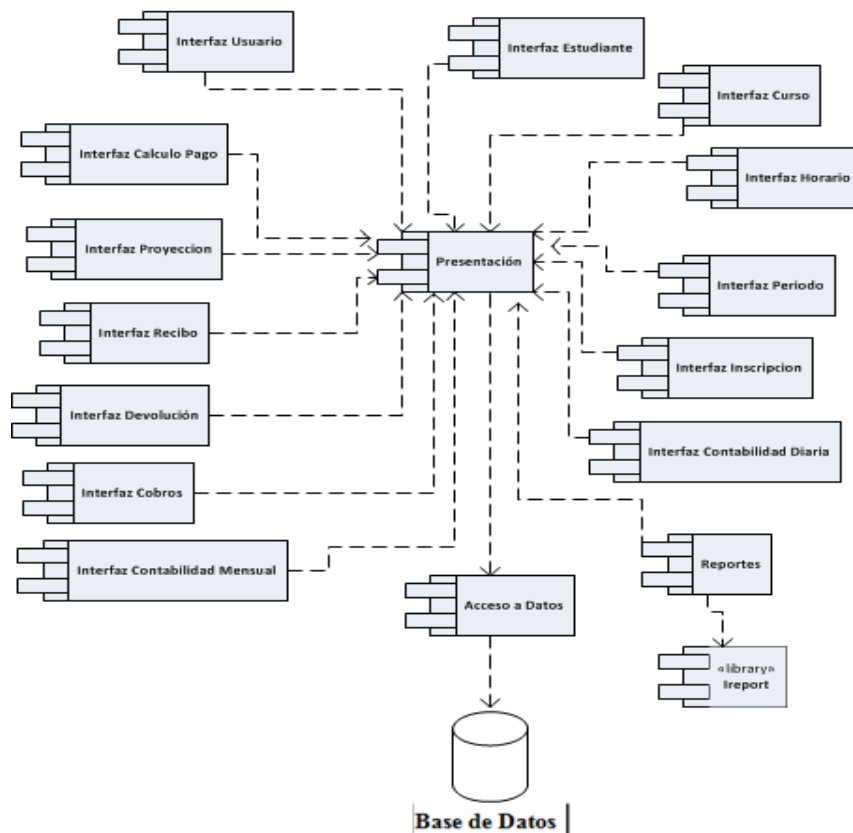


Figura 4-2. Diagrama de Componentes

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.1.1.3 Diagrama de Casos de Usos

Son descripciones de la funcionalidad del negocio, sirven para especificar la funcionalidad y el comportamiento del sistema mediante la interacción con los usuarios.

En la figura 5-2 el usuario administrador ingresa al sistema SGAF por medio de la autenticación, una vez que se loguea tendrá acceso al menú de Gestión Académica, Gestión Financiera y Reportes.

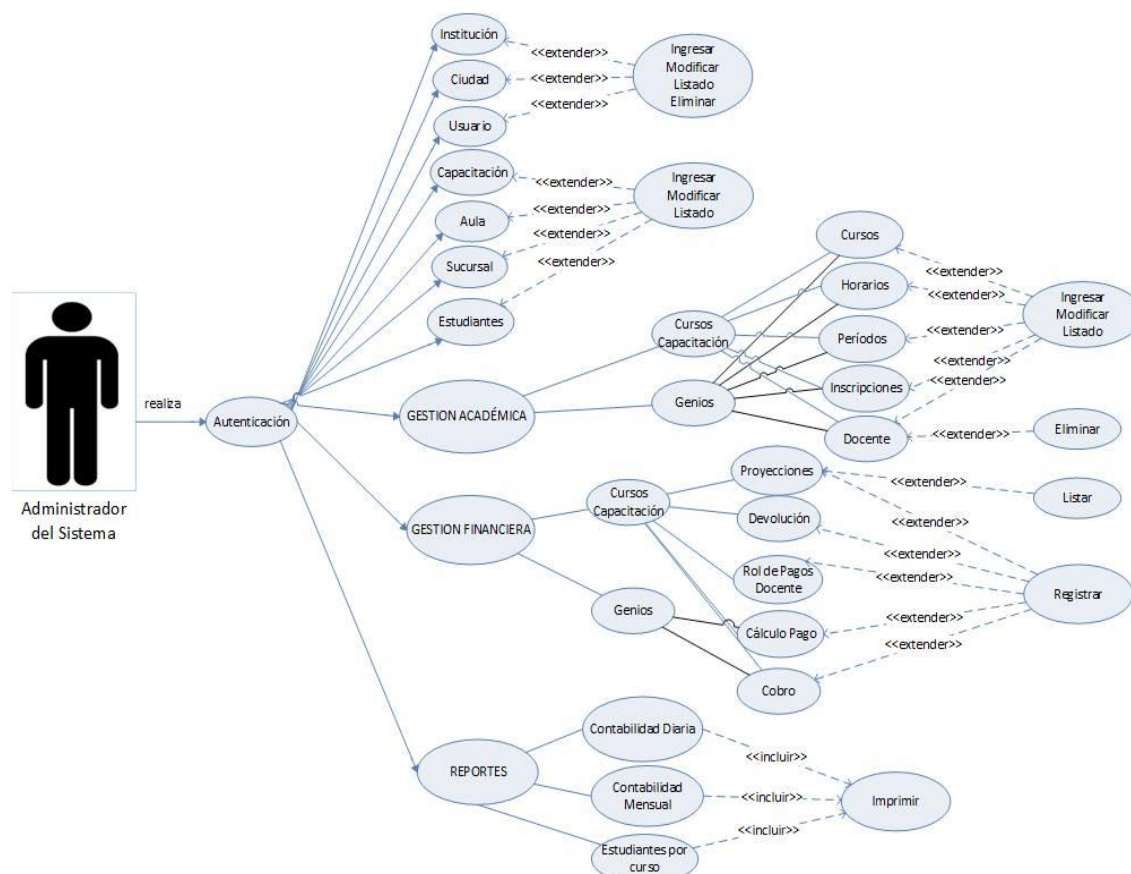


Figura 5-2. Diagrama de Caso de Uso

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.1.2 Tecnologías de desarrollo

HT2: Definición de las tecnologías de desarrollo

La segunda historia técnica está relacionada con la definición de las tecnologías de desarrollo del sistema como se detalla en la Tabla 6-2 la misma que se muestra a continuación:

Tabla 6-2: Tecnologías de Desarrollo

HISTORIA TÉCNICA	
Número: HT2	Sprint Asignado: 1
Tiempo Estimado:	4 horas
Responsable:	Patricia Allauca – Mercy Sani
Descripción: Como administrador del sistema solicito se defina las tecnologías de desarrollo	

Pruebas de Aceptación:

- La definición de las tecnologías requeridas para el desarrollo del sistema tienen que ser definidas por los miembros del equipo de trabajo.

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Las tecnologías utilizadas para el desarrollo de la aplicación web son:

- Base de Datos → PostgreSQL 9.4
- Lenguaje de Programación → Java
- Framework de persistencia → JPA
- Framework UI → Primefaces

2.5.2.1.3 Estándar de Codificación

HT3: Definición del Estándar de Codificación

Esta historia de usuario se relaciona con la definición del Estándar de Codificación para el desarrollo del sistema y se detalla en la Tabla 7-2:

Tabla 7-2: Estándar de Codificación

HISTORIA TÉCNICA	
Número: HT3	Sprint Asignado: 1
Tiempo Estimado:	4 horas
Responsable:	Patricia Allauca – Mercy Sani
Descripción: Como administrador del sistema solicito se defina el Estándar de Codificación	
Pruebas de Aceptación: <ul style="list-style-type: none"> - La definición del Estándar de Codificación para el desarrollo del sistema tiene que ser definido por los miembros del equipo de trabajo. 	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

El estándar de codificación para la base de datos se describe de la siguiente manera:

Tabla.- si está conformada por una o varias palabras se escriben con minúscula y seguido por ejemplo calculopago.

Campos.- los campos que coincidan con los atributos de otra tabla se le añadirá el guión bajo junto con el nombre de la tabla por ejemplo celular_docente, celular_estudiante, los demás campos se escribirán con sus respectivos nombres y en minúsculas.

La Pk puede empezar con la sílaba id seguido de un guión bajo y el nombre de la tabla por ejemplo en la tabla estudiante sería id_estudiante.

Clave Primaria.- se la representa con la sílaba pk seguido de un guión bajo y el nombre de la tabla por ejemplo pk_estudiante.

Clave Foránea.- se la representa con la sílaba fk seguido de un guión bajo y el nombre de las tablas relacionadas separadas por un guión bajo por ejemplo: fk_aula_calculopago.

Funciones.- el nombre de las funciones creadas siempre empieza con la letra f seguido de un guión bajo, la acción que va a realizar seguido de un guión bajo y el nombre de la función por ejemplo: f_insertar_calculo_pago.

El estándar de codificación para desarrollar la aplicación debe permitir asegurarse de que todos los programadores del proyecto trabajen de forma coordinada.

Clases.- inician con la primera letra en mayúscula

Atributos.- se escriben de la misma manera que los campos de la base de datos.

Métodos.- los métodos creados empiezan con letra minúscula, en el caso que esté formada de dos o más palabras la inicial de la primera palabra va con minúscula y las iniciales de las siguientes palabras van con mayúscula y seguidos por ejemplo: guardarInscripcion.

Formularios.- el nombre de los formularios empiezan con mayúscula.

2.5.2.1.4 Diseño de la Base de Datos

HT4: Diseño de la Base de Datos

Está relacionada con el diseño de la base de datos como se detalla en la Tabla 8-2 la misma que se muestra a continuación:

Tabla 8-2: Diseño de la BD

HISTORIA TÉCNICA	
Número: HT4	Sprint Asignado: 1
Tiempo Estimado:	40 horas
Responsable:	Patricia Allauca – Mercy Sani
Descripción: Como administrador del sistema solicito se defina el Diseño de la Base de Datos	
Pruebas de Aceptación: <ul style="list-style-type: none"> - El diseño de la base de datos para el desarrollo del sistema debe ser realizada por los miembros del equipo de trabajo. 	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

La base de datos se la desarrolló en PostgreSQL, en la Figura 6-2 y Figura 7-2 se puede observar el modelo conceptual y físico respectivamente en los que se visualiza las entidades, atributos y relaciones que se requieren para iniciar con el desarrollo del sistema.

2.5.2.1.4.1 Modelo de Datos Conceptual

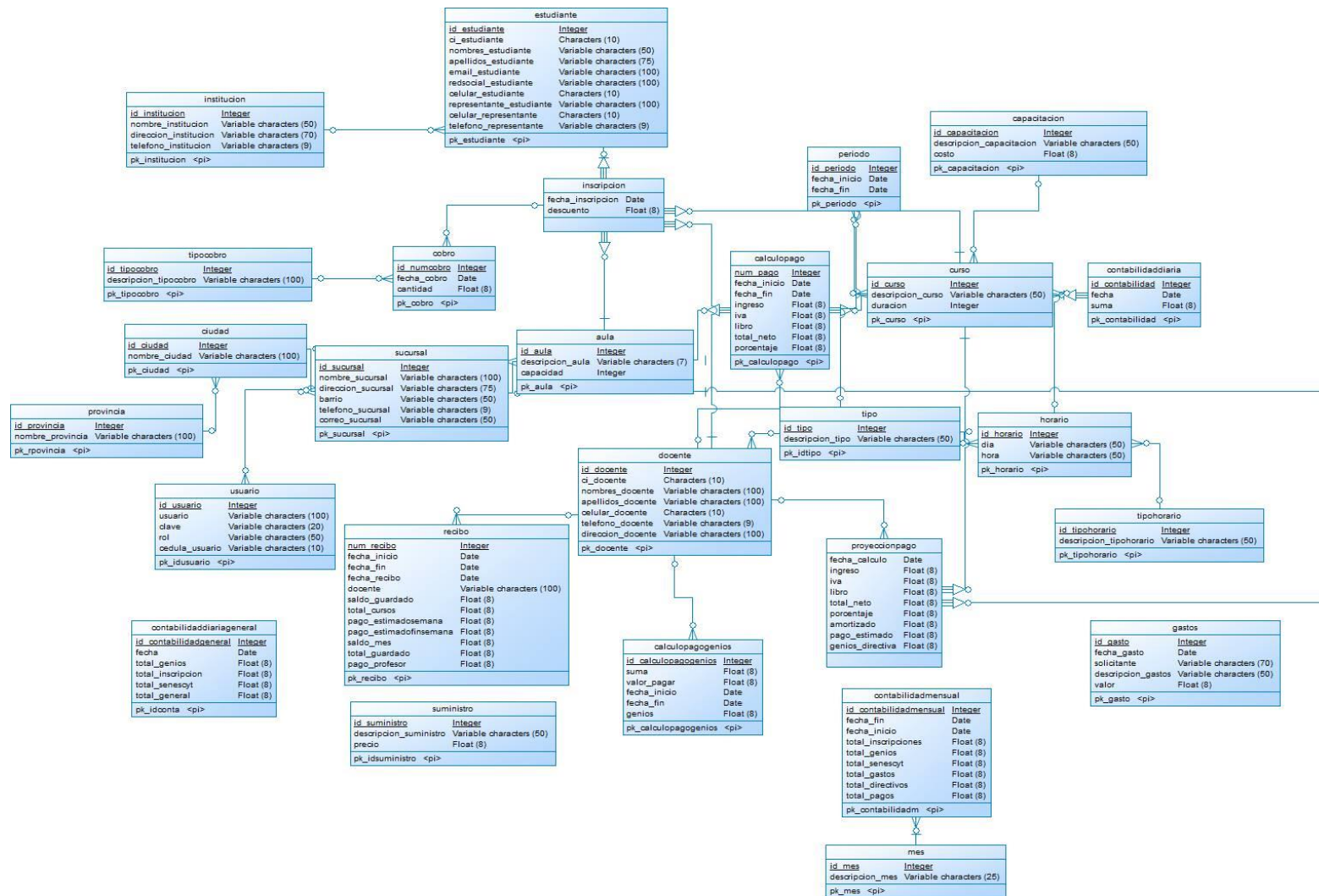


Figura 6-2. Modelo de Datos Conceptual
Realizado por: Allauca Onate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.1.4.2 Modelo de Datos Físico

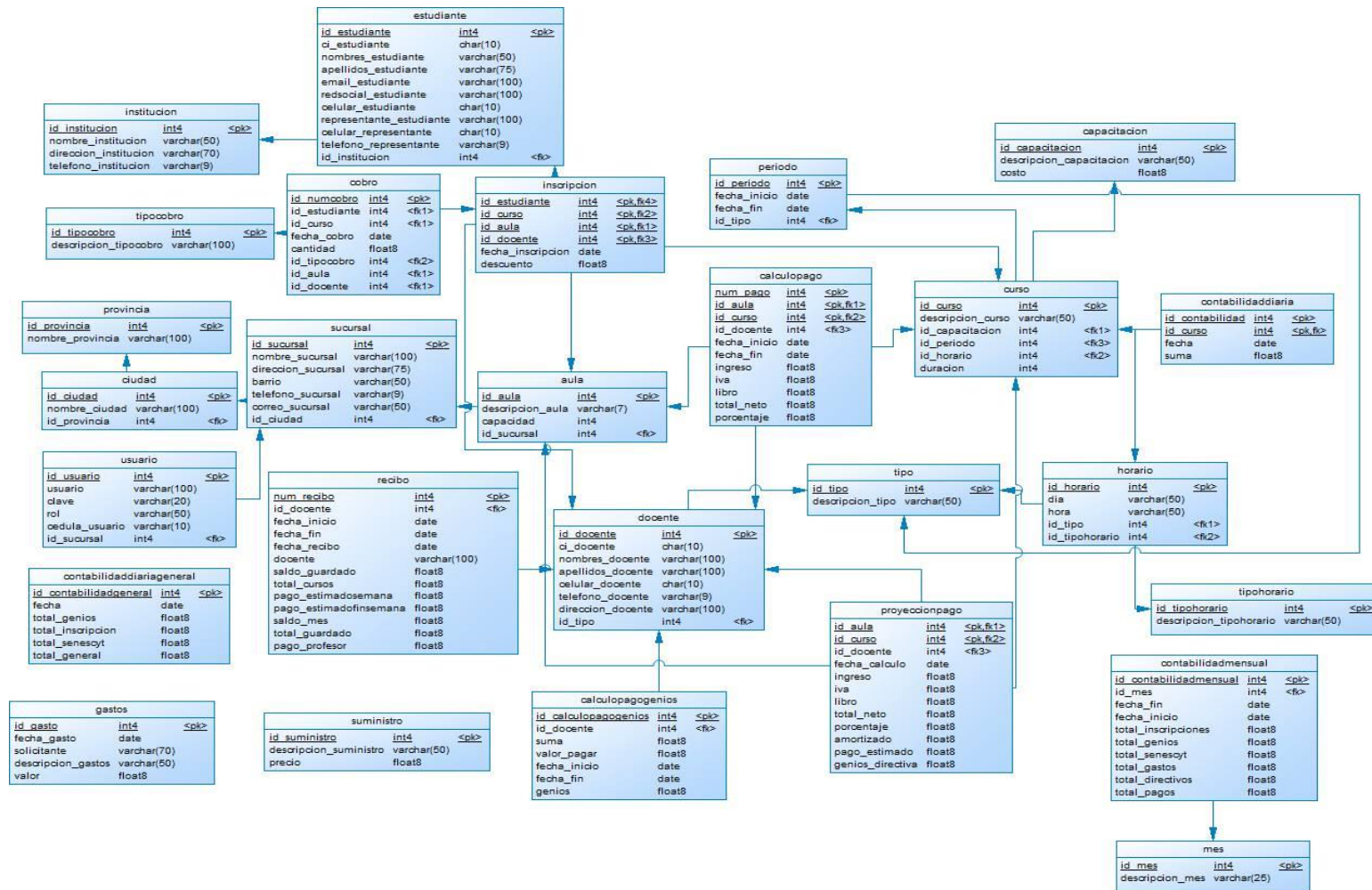


Figura 7-2. Modelo de Datos Físico

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.1.4.3 Diccionario de Datos

Es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa.

La información que se detalla es el nombre de las columnas con sus respectivos tipos de datos, tamaño, claves primarias y foráneas.

El diccionario de datos de todas las tablas se detalla en el Anexo B.

2.5.2.1.5 Interfaz de Usuario

HT5: Definición de la Interfaz de Usuario

La quinta historia técnica está relacionada con la definición de la interfaz de usuario la misma que se detalla a continuación:

Tabla 9-2: Interfaz de Usuario

HISTORIA TÉCNICA	
Número: HT5	Sprint Asignado: 1
Tiempo Estimado:	8 horas
Responsable:	Patricia Allauca – Mercy Sani
Descripción: Como administrador del sistema solicito se defina la Interfaz de Usuario	
Pruebas de Aceptación:	
- La definición de la interfaz de usuario tiene que ser aprobada por parte de la directiva junto con los miembros del equipo de trabajo.	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

La interfaz elegida para la implementación del sistema “SGAF”, contiene los colores representativos de la institución “GENIOS TRABAJANDO” así como también el sello, la ubicación de los enlaces en cada una de las páginas es la misma incluyendo la página principal. Cabe aclarar que “SGAF” tiene la particularidad de permitir que la interfaz de la página principal sea heredada por las demás páginas.

Página Principal

La página principal tiene los siguientes componentes en el encabezado tiene el nombre, sello y el slogan de la institución.

La plantilla en las que se basan las demás páginas está formada por:

- **Encabezado:** contiene el nombre, sello, slogan de la institución, el nombre el usuario que accedió al sistema y el cierre de sesión.
- **Pie:** Al final de todas las páginas se puede visualizar los derechos reservados y el ícono de ayuda.
- **Color:** se utilizó los colores representativos de la institución como son el amarillo, rojo, blanco y negro.

- **Menú:** contiene submenús desplegables de opciones que organizan los principales procesos que se llevan a cabo en la institución, se encuentran en la parte izquierda, como son la Gestión Académica, Gestión Financiera y Reportes, las mismas que serán visualizadas y ejecutadas en la parte central de la página.

Todas las características mencionadas anteriormente relacionadas con la interfaz de usuario se pueden visualizar de una mejor manera en la Figura 8-2 y 9-2.



Figura 8-2. Login SGAF

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016



Figura 9-2. Página Principal

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Generación de Reportes

Para la generación de los reportes, el encabezado contiene el nombre, sello y slogan de la institución, en la parte del contenido se mostrará los datos requeridos para cada reporte, además de las firmas de responsabilidad sea de la secretaria, docentes o a su vez de los directivos de dicha institución. Se puede visualizar de una mejor manera en la Figura 10 -2.

Recibo Pago Docente


Centro de Capacitación Académica y Pre-Universitaria		
“GENIOS TRABAJANDO”		
ROL DE PAGOS		
Riobamba, Sábado 31 de Octubre del 2015		N° 6
Docente: Diego Fernando Mayorga Pérez		
Del: Lunes 10 de Agosto del 2015	Al: Sábado 31 de Octubre del 2015	
Curso	Fecha Inicio	Fecha Fin
Sábados 8H00 a 12H30	17 de Octubre del 2015	26 de Marzo del 2016
Lunes a Jueves 17H00 a 19H00	19 de Octubre del 2015	31 de Marzo del 2016
		Total
		203,04
		266,32
		Total a Pagar: 469,36
FIRMA DOCENTE		FIRMA GERENTE

Figura 10-2. Recibo Pago Docente

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

2.5.2.2 Sprint 2

HU11: Registrar las inscripciones de los estudiantes

La historia de usuario 11 está relacionada con el registro de las inscripciones de los estudiantes y se detalla a continuación:

Tabla 10-2: Historia de Usuario HU11

HISTORIA DE USUARIO	
Número: 11	Sprint Asignado: 2
Tiempo Estimado:	40 horas
Responsable:	Patricia Allauca
Descripción: Como director requiero registrar las inscripciones de los estudiantes	
Tarea 1: Creación de la función e implementación del método que permita realizar el registro de inscripciones <ul style="list-style-type: none">- Implementar la función <code>f_insertar_inscripcion_senescyt</code>, el mismo que necesita 4 parámetros de entrada que son: estudiante, curso, fecha y descuento.- Insertar los datos en la tabla.- Implementar el método denominado <code>InsertarInscripcion</code>.	
Tarea 2: Diseño de la interfaz de usuario para realizar el registro de inscripciones <ul style="list-style-type: none">- Diseñar la página para buscar al estudiante ya sea por nombres o por cédula.- Método que permita visualizar el listado de estudiantes de la base de datos denominado <code>cargarEstudiantesPorNombreoCedula</code>.- Diseñar un diálogo que permita realizar el registro de inscripciones.- Generar botones tanto para el Guardar como para el Cancelar.	
Pruebas de Aceptación: <ul style="list-style-type: none">- Mensaje de confirmación al momento de realizar el registro.- Verificar la función para realizar el registro de las inscripciones de los estudiantes.	

Realizado por: Patricia Allauca

Tabla 11-2: Tarea de Ingeniería 1 HU11

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Registrar las inscripciones de los estudiantes	
Nombre de la Tarea: Creación de la función e implementación del método que permita realizar el registro de inscripciones.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Creación de la función que permita registrar las inscripciones de los estudiantes.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la función para realizar el registro de las inscripciones de los estudiantes. 	

Realizado por: Patricia Allauca

Se procede a verificar las pruebas de aceptación de las diferentes historias de usuario para así determinar si satisfacen las funcionalidades del sistema.

A continuación se detalla la prueba de aceptación de la historia de usuario 11 relacionada con el registro de inscripciones y se muestra en las Tabla 12-2.

Tabla 12-2: Prueba de Aceptación 1 HU11

PRUEBA DE ACEPTACIÓN	
Código: HU11.1	Nombre Historia de Usuario: Registrar las inscripciones de los estudiantes.
Nombre de la Prueba: Mensaje de confirmación al momento de realizar el registro.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de registrar las inscripciones es necesario mostrar un mensaje de confirmación.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El id del estudiante→8, id del curso→39 no deben estar almacenado en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Seleccionar el curso (39) - Llenar la fecha('2015-09-07') - Dar clic en el botón Guardar. - Se muestra el mensaje correspondiente. 	
Resultado Esperado: <ul style="list-style-type: none"> - Los datos son almacenados correctamente en la tabla inscripción en la base de datos. - Al guardar la información deberá mostrar un mensaje de confirmación "Ingreso correcto". - Cuando no se pueda registrar deberá mostrar un mensaje de error "No se puede registrar". 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

En el Anexo C se mostrarán las historias de usuario restantes con las tareas de ingeniería y pruebas de aceptación.

2.5.2.2.1 Avance del Proyecto

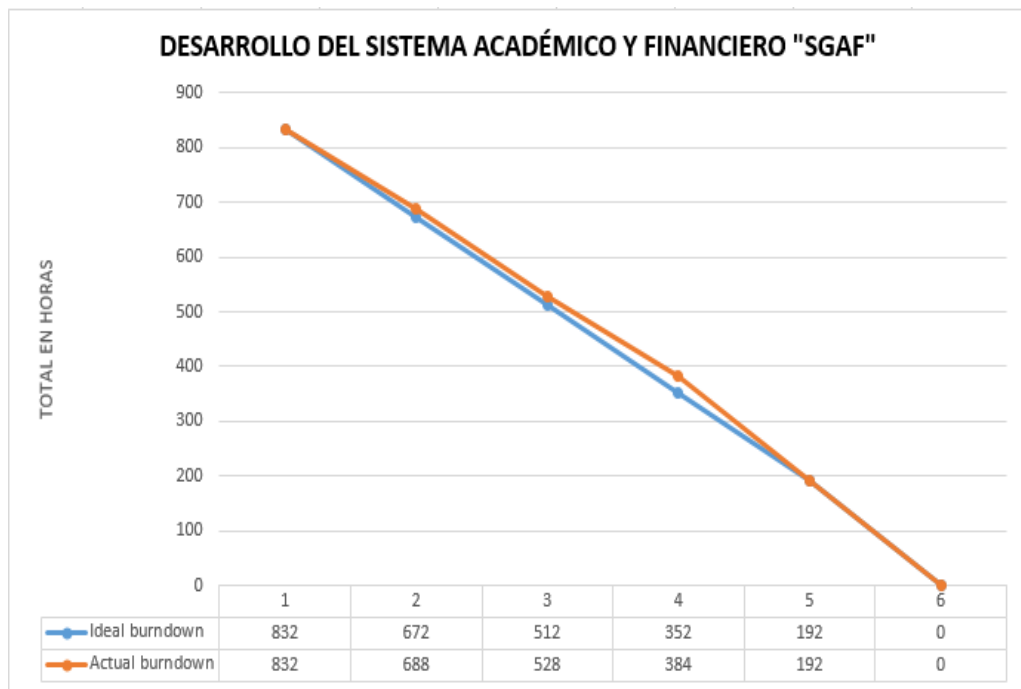


Figura 11-2. BurnDownChart SGAF

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

NOTA: Como se puede apreciar en la Figura 11-2, cada uno de los puntos representa un sprint, se puede deducir que según la línea de progreso actual, nuestro proyecto necesitó algunas horas adicionales en relación al tiempo estimado para culminar con los Sprint 2,3 y el Sprint 4 requirió de un mayor número de horas, pero en el Sprint 5 los tiempos se fueron ajustando hasta cumplir con el objetivo de terminar con el proyecto en la fecha establecida.

CAPÍTULO III

3 MARCO DE RESULTADOS Y DISCUSIÓN DE RESULTADOS

De acuerdo a las necesidades del usuario consideradas para el desarrollo del sistema web es necesario establecer los mecanismos que permitan la obtención de los datos y el análisis de resultados.

En el trabajo de titulación la información se obtuvo mediante la técnica de recopilación de información la Observación con su instrumento Ficha de observación, la misma que se realizó mediante la toma de tiempos que se demoran en realizar los principales procesos y la cantidad de errores que se presentan en los mismos. También otra técnica que se empleó es la Encuesta con su instrumento el Cuestionario, el cual permitirá medir la usabilidad del sistema y se la realizó a las 3 secretarías y 2 directivos que manejarán el sistema. Con los resultados obtenidos se podrá determinar qué cambios o mejoras se presentaron en el desarrollo de la aplicación web.

3.1 Técnicas utilizadas para el análisis y obtención de resultados

3.1.1 Observación

Esta técnica fue empleada mediante el instrumento de la Ficha de Observación para poder realizar un análisis comparativo de los tiempos que se emplean en realizar los principales procesos tanto manualmente como a través del sistema “SGAF”, de la misma manera la cantidad de errores encontrados.

3.1.1.1 Fichas de Observación con respecto al tiempo

A continuación se detallan las fichas de observación relacionadas al tiempo de respuesta, los mismos que fueron medidos a través de un cronómetro.

En la tabla 1-3 se muestra la ficha de observación de la toma de los tiempos de los principales procesos sin la utilización del sistema.

Institución: Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”

Tema: Medición del tiempo que se emplea para realizar los principales procesos de forma manual en la institución.

Lugar: Veloz entre Pichincha y García Moreno.

Nombres del investigador(es):

- Mercy Sani
- Patricia Allauca

Fuente: Mabel Ocaña

Fecha de la observación: 23 de Mayo del 2016

FICHA DE OBSERVACIÓN (Tiempo sin Sistema)

Tabla 1-3: Ficha de Observación (Tiempo sin Sistema)

Procesos	Unidad de medida	Tiempo empleado
¿Cuál es el tiempo empleado en el registro de la inscripción de un estudiante?	minutos y segundos	6 minutos y 29 segundos
¿Cuánto tiempo se demora en conocer el valor que adeuda un estudiante?	minutos y segundos	1 minuto y 40 segundos
¿Cuánto tiempo se tarda en registrar y generar los recibos de los pagos que realizan los estudiantes de un determinado curso?	minutos y segundos	10 minutos y 44 segundos
¿Cuánto tiempo se demora en realizar todos los cálculos necesarios para poder pagar al docente?	minutos y segundos	26 minutos y 57 segundos
¿Cuánto tiempo se demora en generar el reporte de la contabilidad mensual?	minutos y segundos	8 minutos y 43 segundos

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la tabla 2-3 se muestra la ficha de observación de la toma de los tiempos de los principales procesos con la utilización del sistema.

Institución: Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”

Tema: Medición del tiempo que se emplea para realizar los principales procesos con el sistema web en la institución.

Lugar: Veloz entre Pichincha y García Moreno.

Nombres del investigador(es):

- Mercy Sani
- Patricia Allauca

Fuente: Mabel Ocaña

Fecha de la observación: 23 de Mayo del 2016

FICHA DE OBSERVACIÓN (Tiempo con Sistema)

Tabla 2-3: Ficha de Observación (Tiempo con Sistema)

Procesos	Unidad de medida	Tiempo empleado
¿Cuál es el tiempo empleado en el registro de la inscripción de un estudiante?	minutos y segundos	3 minutos y 10 segundos
¿Cuánto tiempo se demora en conocer el valor que aún adeuda un estudiante?	minutos y segundos	48 segundos
¿Cuánto tiempo se tarda en registrar y generar los recibos de los pagos que realizan los estudiantes de un determinado curso?	minutos y segundos	2 minutos y 23 segundos
¿Cuánto tiempo se demora en realizar todos los cálculos necesarios para poder pagar al docente?	minutos y segundos	3 minutos y 44 segundos
¿Cuánto tiempo se demora en generar el reporte de la contabilidad mensual?	minutos y segundos	1 minuto y 8 segundos

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Se procede a realizar la tabulación de los mismos, analizando las fichas de observación con respecto al tiempo para posteriormente presentarlos en un gráfico de diagrama de barras con el respectivo análisis.

En la tabla 3-3 se detalla los resultados de la toma de tiempos del proceso 1

Tabla 3-3: Proceso 1

Proceso 1. ¿Cuál es el tiempo empleado en el registro de la inscripción de un estudiante?	
	Tiempo
Tiempo empleado sin sistema	0:06:29
Tiempo empleado con sistema	0:03:10

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 1-3 se muestra los datos obtenidos del proceso 1.

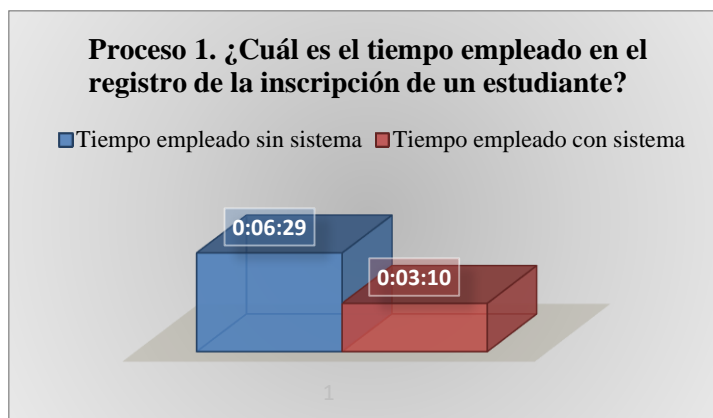


Figura 1-3. Resultados Proceso 1

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

En el proceso 1 relacionado con el registro de la inscripción de un estudiante en la ficha de observación “Tabla 1-3” se puede visualizar que el tiempo empleado sin la utilización del sistema es de 6 minutos con 29 segundos, mientras que con la utilización del sistema “Tabla 2-3” es de 3 minutos con 10 segundos, se concluye que existe una diferencia de 3 minutos y 19 segundos entre los dos sistemas.

En la tabla 4-3 se detalla los resultados de la toma de tiempos del proceso 2.

Tabla 4-3: Proceso 2

Proceso 2. ¿Cuánto tiempo se demora en conocer el valor que aún adeuda un estudiante?	
	Tiempo
Tiempo empleado sin sistema	0:01:40
Tiempo empleado con sistema	0:00:48

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 2-3 se muestra los datos obtenidos del proceso 2.

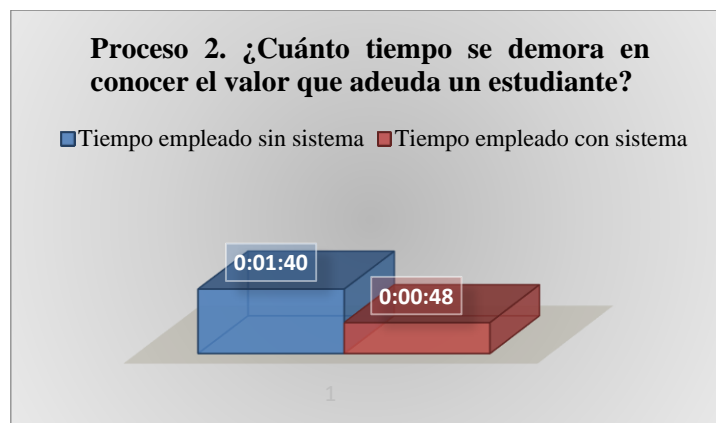


Figura 2-3. Resultados Proceso 2

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

En el proceso 2 relacionado con el valor que adeuda un estudiante en la ficha de observación “Tabla 1-3” se puede visualizar que el tiempo empleado sin la utilización del sistema es de 1 minuto con 40 segundos, mientras que con la utilización del sistema “Tabla 2-3” es de 48 segundos, se concluye que existe una diferencia de 52 segundos entre los dos sistemas.

En la tabla 5-3 se detalla los resultados de la toma de tiempos del proceso 3

Tabla 5-3: Proceso 3

Proceso 3. ¿Cuánto tiempo se tarda en registrar y generar los recibos de los pagos que realizan los estudiantes de un determinado curso?	
	Tiempo
Tiempo empleado sin sistema	0:10:44
Tiempo empleado con sistema	0:02:23

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 3-3 se muestra los datos obtenidos del proceso 3.

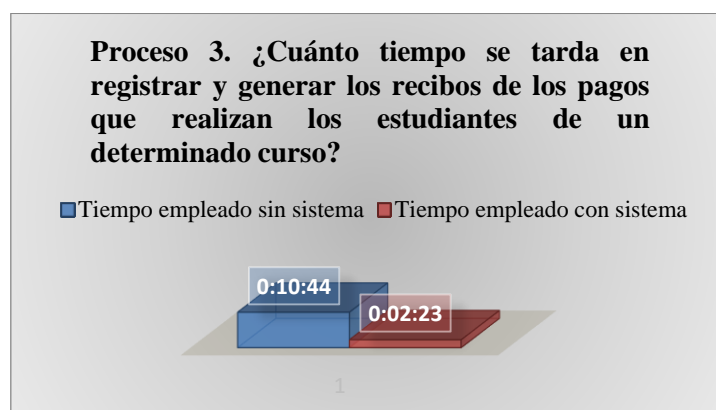


Figura 3-3. Resultados Proceso 3

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

En el proceso 3 relacionado con el registro y generación de los recibos de los pagos que realizan los estudiantes, en la ficha de observación “Tabla 1-3” se puede visualizar que el tiempo empleado sin la utilización del sistema es de 10 minutos con 44 segundos, mientras que con la

utilización del sistema “Tabla 2-3” es de 2 minutos con 23 segundos, se concluye que existe una diferencia de 8 minutos con 21 segundos entre los dos sistemas.

En la tabla 6-3 se detalla los resultados de la toma de tiempos del proceso 4

Tabla 6-3: Proceso 4

Proceso 4. ¿Cuánto tiempo se demora en realizar todos los cálculos necesarios para poder pagar al docente?	
	Tiempo
Tiempo empleado sin sistema	0:26:57
Tiempo empleado con sistema	0:03:44

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 4-3 se muestra los datos obtenidos del proceso 4.

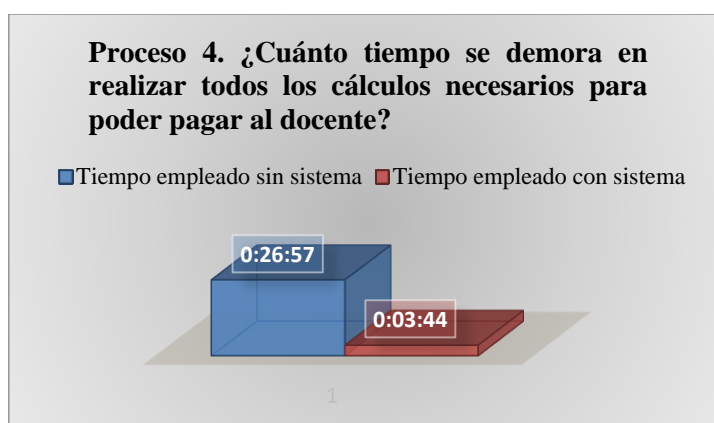


Figura 4-3. Resultados Proceso 4

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

En el proceso 4 relacionado con la realización de los cálculos necesarios para poder pagar al docente, en la ficha de observación “Tabla 1-3” se puede visualizar que el tiempo empleado sin la utilización del sistema es de 26 minutos con 57 segundos, mientras que con la utilización del sistema “Tabla 2-3” es de 3 minutos con 44 segundos, se concluye que existe una diferencia de 23 minutos con 13 segundos entre los dos sistemas.

En la tabla 7-3 se detalla los resultados de la toma de tiempos del proceso 5

Tabla 7-3: Proceso 5

Proceso 5. ¿Cuánto tiempo se demora en generar el reporte de la contabilidad mensual?	
	Tiempo
Tiempo empleado sin sistema	0:08:43
Tiempo empleado con sistema	0:01:08

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 5-3 se muestra los datos obtenidos del proceso 5.

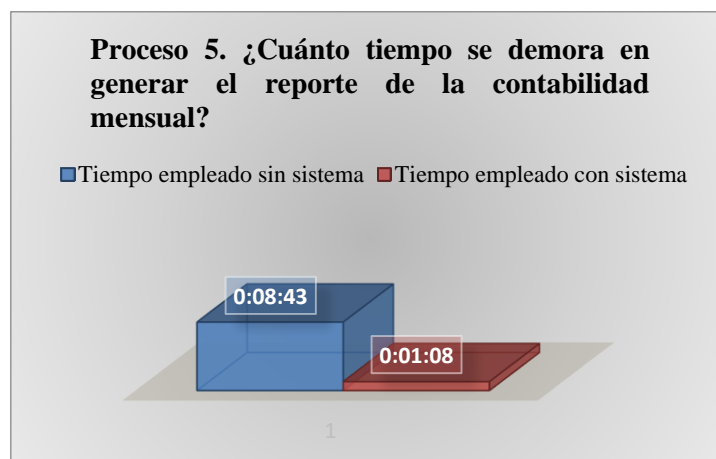


Figura 5-3. Resultados Proceso 5

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

En el proceso 5 relacionado con la generación de la contabilidad mensual, en la ficha de observación “Tabla 1-3” se puede visualizar que el tiempo empleado sin la utilización del sistema es de 8 minutos con 43 segundos, mientras que con la utilización del sistema “Tabla 2-3” es de 1 minutos con 8 segundos, se concluye que existe una diferencia de 7 minutos con 35 segundos entre los dos sistemas.

A continuación en la tabla 8-3 se muestra el análisis comparativo entre las 2 fichas de observación correspondientes al tiempo utilizado.

3.1.1.1.1 Análisis Comparativo de la toma de Tiempos

Tabla 8-3: Análisis Comparativo Tiempo

Proceso	Sin el sistema	Con el sistema	Resultado
¿Cuál es el tiempo empleado en el registro de la inscripción de un estudiante?	6 minutos y 29 segundos	3 minutos y 10 segundos	Con el sistema el tiempo de respuesta que transcurre en el registro de la inscripción de un estudiante se redujo en 3 minutos con 19 segundos.
¿Cuánto tiempo se demora en conocer el valor que adeuda un estudiante?	1 minuto y 40 segundos	48 segundos	Con el sistema el tiempo de respuesta que se usa para conocer el valor que adeuda un estudiante se redujo en total de 52 segundos.

¿Cuánto tiempo se tarda en registrar y generar los recibos de los pagos que realizan los estudiantes de un determinado curso?	10 minutos y 44 segundos	2 minutos y 23 segundos	Con el sistema el tiempo de respuesta utilizado para el registro y generación de los recibos de los pagos de los estudiantes se redujo en un total de 8 minutos con 21 segundos.
¿Cuánto tiempo se demora en realizar todos los cálculos necesarios para poder pagar al docente?	26 minutos y 57 segundos	3 minutos y 44 segundos	Con el sistema el tiempo de respuesta empleado para realizar los cálculos necesarios para poder pagar al docente se redujo en un total de 23 minutos con 13 segundos.
¿Cuánto tiempo se demora en generar el reporte de la contabilidad mensual?	8 minutos y 43 segundos	1 minuto y 8 segundos	Con el sistema el tiempo de respuesta usado para generar el reporte de la contabilidad mensual se redujo en un total de 7 minutos con 35 segundos.
Totales Tiempo	54 minutos con 33 segundos	11 minutos con 13 segundos	Con el sistema el tiempo de respuesta empleado en los principales procesos que maneja la institución se redujo en un total de 43 minutos con 20 segundos.

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

3.1.1.2 Fichas de Observación con respecto a la cantidad de errores

En la tabla 9-3 se muestra la ficha de observación con la cantidad de errores encontrados sin la utilización del sistema.

Institución: Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”

Tema: Cantidad de errores encontrados en la realización de los procesos de forma manual en la institución.

Lugar: Veloz entre Pichincha y García Moreno.

Nombres del investigador(es):

- Mercy Sani
- Patricia Allauca

Fuente: Mabel Ocaña

Fecha de la observación: 20 de Mayo del 2016

FICHA DE OBSERVACIÓN (Cantidad de errores encontrados sin el sistema)

Tabla 9-3: Ficha de Observación (Cantidad de errores encontrados sin el sistema)

Procesos	Nº	Descripción del error	Observación
Capacitación			No existen errores
Estudiante	1	No existe toda la información de los datos personales de los estudiantes.	
	2	No existe un identificador que permita distinguir a cada estudiante.	
Cursos con sus respectivos horarios, periodos y aulas			No existen errores
Docentes			No existen errores
Inscripciones	1	No existe el acceso inmediato a la información o datos de los estudiantes, cursos, horarios, periodos, capacitaciones para realizar la inscripción	
	2	No existe el control adecuado al momento de registrar a los estudiantes puesto que sobrepasan la capacidad disponible que tiene un determinado curso provocando que falte uno o varios pupitres o lo que es lo mismo el o los estudiantes deben estar de pie recibiendo clases.	
	3	Confusión al tener aparentemente al mismo estudiante en dos cursos, esto se debe a que la poca información que se registra no es suficiente para poder distinguirlos, por ejemplo estudiantes con los mismos nombres y apellidos.	
	4	Cuando los datos personales del estudiante requieren ser cambiados se los tacha de la hoja de inscripción o se los pone con lápiz a lado, lo que provoca confusión al no conocer cuál es la información correcta o actual.	
Cambio Curso	1	No controlan la capacidad disponible que tienen los demás cursos para poder realizar el cambio, provocando que sobrepase la capacidad del mismo.	
	2	No tienen la información oportuna de los cursos a los que se puede cambiar el estudiante provocando pérdida de tiempo para la persona que atiende y para la persona que recibe el servicio.	
Cobros	1	No registrar en algunas ocasiones las fechas de los pagos que realizan los estudiantes, registrar con lápiz o con esfero en hojas provocando que dicha información no sea clara.	
	2	Inconformidad al no conocer cuánto ha pagado, cuanto le falta por pagar y cuanto de descuento tienen los estudiantes.	
	3	Registro de pagos de un mismo estudiante en varios cursos, esto ocurre porque no se sabe si es la misma persona o es otra con los	

		mismos nombres o apellidos.	
Devoluciones	1	El devolver el dinero sea de la inscripción o de las cuotas provoca desajustes en los cálculos realizados, puesto que en los mismos se toma en cuenta dinero ya no existente.	
Proyecciones	1	No existe un control adecuado sobre el número de estudiantes retirados, puesto que las proyecciones se las realiza en base a los estudiantes inscritos.	
	2	Confusión al realizar los cálculos del iva, libros.	
	3	No se toma en cuenta de forma correcta los descuentos que se realizan para las proyecciones de los cursos.	
Cálculo Pago Docentes	1	No existe un control sobre los saldos guardados que tiene cada uno de los docentes.	
	2	No existe un control adecuado sobre el número de estudiantes retirados, lo cual provoca errores en el cálculo de los valores de los libros y del iva de esta forma los valores reales no son iguales a los calculados.	
	3	Una vez que se ha calculado el dinero a pagar a los docentes a través de la fórmula, existe confusión al momento de multiplicarlo por un determinado porcentaje puesto que les toca buscar entre las hojas si son cursos que se dictan en la semana o a su vez los fines de semana puesto que entre ellos cambia dicho porcentaje, lo que provocaría que algunos docentes se les aumente o disminuya el valor correcto que deben cobrar.	
	4	Confusión al momento de sumar las cuotas de los estudiantes de todos los cursos a los que el docente imparte las clases.	
Contabilidad Diaria	1	No se lleva el registro de los gastos diarios al momento de generar la contabilidad diaria.	
Contabilidad Mensual	1	No existe el control de los ingresos y egresos con los que realmente cuenta la institución, provocando que se desconozca si la misma obtiene utilidades o pérdidas.	

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Se procede a analizar la ficha de observación con respecto a la cantidad de errores encontrados en los procesos que se llevan a cabo sin la utilización del sistema mediante un gráfico de diagrama pastel.

En la tabla 10-3 se detalla el número de procesos que contienen errores y aquellos que no tienen.

Tabla 10-3: Procesos realizados sin la utilización del sistema

Número de procesos que tienen errores y números de procesos que no tienen, realizados sin la utilización del sistema		
	Nº de Procesos	%

Con Errores	9	75%
Sin errores	3	25%
TOTAL	12	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 6-3 se muestra los datos obtenidos de la tabla 10-3.

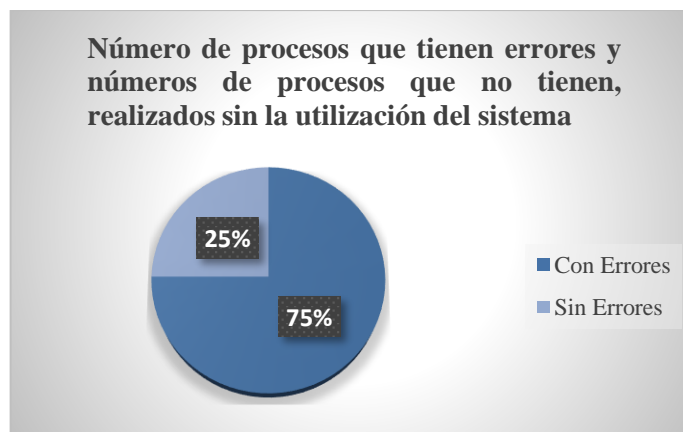


Figura 6-3. Procesos realizados sin la utilización del sistema

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis de Resultados de la cantidad de errores encontrados sin la utilización del sistema.

Los procesos que se realizan en la institución son un total de 12, en la ficha de observación “Tabla 9-3” se puede visualizar que 9 procesos tienen errores lo que correspondería al 75%, mientras que los procesos restantes no poseen errores lo que correspondería al 25%, se concluye que sin la utilización del sistema la mayoría de los procesos realizados contienen errores.

En los 9 procesos existe un total de 21 errores, en la “Tabla 11-3” se detalla el número de errores encontrados en cada uno de los 9 procesos así como el porcentaje que representa cada uno de ellos sobre el 75% (procesos que tienen errores) y no del 100% porque el 25% de los procesos que se manejan no contienen errores.

Tabla 11-3: Número de errores por cada proceso

Proceso	N° de error	Porcentaje
Estudiantes	2	7.14%
Inscripciones	4	14.29%
Cambio de Cursos	2	7.14%
Cobros	3	10.71%
Devoluciones	1	3.57%
Proyecciones	3	10.71%
Cálculo Pago	4	14.29%
Contabilidad Diaria	1	3.57%
Contabilidad Mensual	1	3.57%
Totales	21	75%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la tabla 12-3 se muestra la ficha de observación con la cantidad de errores encontrados con la utilización del sistema.

Institución: Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”

Tema: Cantidad de errores encontrados en la realización de los procesos mediante la utilización del sistema web en la institución.

Lugar: Veloz entre Pichincha y García Moreno

Nombres del investigador(es):

- Mercy Sani
- Patricia Allauca

Fuente: Mabel Ocaña

Fecha de la observación: 20 de Mayo del 2016

FICHA DE OBSERVACIÓN (Cantidad de errores encontrados con el sistema)

Tabla 12-3: Ficha de Observación (Cantidad de errores encontrados con el sistema)

Procesos	Nº	Descripción del error	Observación
Capacitación	1	Al momento de editar el costo de la capacitación el formato utilizado para el mismo no era el correcto puesto que alteraba el valor real ya registrado.	El error fue encontrado dentro de la etiqueta de validación para lo cual se cambió al formato adecuado y ya se solucionó.
Estudiante			No existen errores
Cursos con sus respectivos horarios, periodos y aulas			No existen errores
Docentes			No existen errores
Inscripciones			No existen errores
Cambio Curso			No existen errores
Cobros	1	Al momento de realizar el cobro a los estudiantes se pudo evidenciar que no se cargaban todos los datos.	El error fue encontrado en el método en la sentencia sql puesto que faltaba referenciar a un campo que estaba siendo utilizado y se solucionó.
Devoluciones			No existen errores
Proyecciones	1	Error en la función proyeccionpago por el valor del iva.	El error fue encontrado en la función para la proyeccionpago a los docentes puesto que el iva era constante (12%) sin embargo debió ser parametrizado, es decir se incorporó en la función un nuevo parámetro para que el usuario pueda ingresar el valor del iva y se solucionó.
Cálculo Pago Docentes	1	Error en la función calculopago por el valor del iva.	El error fue encontrado en la función para el cálculo pago a los docentes puesto que el iva era constante (12%) sin embargo debió ser parametrizado, es decir se

			incorporó en la función un nuevo parámetro para que el usuario pueda ingresar el valor del iva y ya fué solucionado.
Contabilidad Diaria			No existen errores
Contabilidad Mensual	1	Al momento de generar los reportes de la contabilidad mensual en otra máquina no se podían visualizar, esto se debió a que la ruta que tenían los subreportes eran absolutas.	El error fue encontrado dentro del reporte cuando se llamaba a los subreportes, lo que se realizó es cambiar la dirección con una ruta relativa y fue solucionado

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis de Resultados de la cantidad de errores con la utilización del sistema

En la ficha de observación “Tabla 12-3” se puede visualizar que existió un total de 5 errores en los procesos realizados con la utilización del sistema, se concluye que existieron errores a nivel de codificación pero fueron solucionados en el momento de ser detectados.

Las fichas de observación de los tiempos de respuesta y la cantidad de errores se detallan en los anexos D y E respectivamente.

3.1.2 Encuesta

La encuesta se la realizó a 5 personas ya que son las encargadas de realizar los procesos que se llevan a cabo en la institución de forma manual, formada por 2 directivos y 3 secretarías.

A continuación en la Tabla 13-3 se detalla el nombre y el cargo que ocupan las personas encuestadas que laboran en la institución:

Tabla 13-3: Personas Encuestadas

Nombres	Cargo
Ing. Diego Mayorga	Director General
Ing. Fausto Hidrobo	Representante Legal
Mabel Ocaña	Secretaria Senescyt
Katherine Granizo	Secretaria Senescyt
Alexandra Medina	Secretaria Genios

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Se realiza la encuesta para medir la usabilidad del sistema “SGAF” para lo cual se procede a realizar la tabulación de los datos, analizando el cuestionario para posteriormente presentarlos en un gráfico de diagrama pastel con el respectivo análisis.

La encuesta realizada a los empleados y directivos se detalla en el anexo F.

A continuación en la tabla 14-3 se detalla los resultados de la pregunta 1.

Tabla 14-3: Pregunta 1

Pregunta 1. ¿Existe un manual de ayuda que explique cómo utilizar el sistema "SGAF"?		
	Valor	Porcentaje
SI	5	100%
NO	0	0%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 7-3 se muestra los datos obtenidos de la pregunta 1.

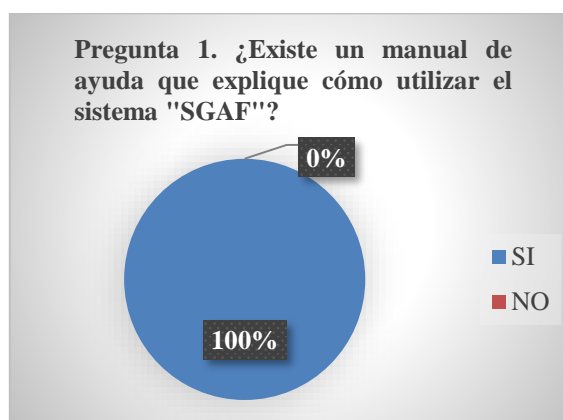


Figura 7-3. Resultados Pregunta 1

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 100% de las personas encuestadas afirman que si existe un manual de ayuda que explica el funcionamiento del sistema “SGAF”.

A continuación en la tabla 15-3 se detalla los resultados de la pregunta 2.

Tabla 15-3: Pregunta 2

Pregunta 2. ¿El recorrido que se hace por el contenido del sistema es fácil?		
	Valor	Porcentaje
SI	4	80%
NO	1	20%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 8-3 se muestra los datos obtenidos de la pregunta 2.



Figura 8-3. Resultados Pregunta 2
Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 80% de las personas encuestadas opinaron que les resulta fácil el recorrido que hacen por el contenido del sistema, en cambio que el 20% afirma que no les resulta fácil, se concluye que la mayor parte de encuestados recorren fácilmente por el contenido del sistema.

A continuación en la tabla 16-3 se detalla los resultados de la pregunta 3.

Tabla 16-3: Pregunta 3

Pregunta 3. ¿La presentación del contenido del sistema como el tipo, tamaño de fuente, textos, color, imágenes y botones siguen el mismo estándar?		
	Valor	Porcentaje
SI	3	60%
NO	2	40%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 9-3 se muestra los datos obtenidos de la pregunta 3.

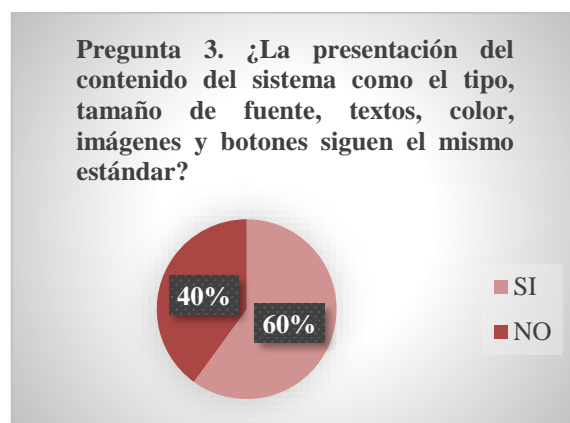


Figura 9-3. Resultados Pregunta 3
Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 60% de las personas encuestadas opinaron que la presentación del contenido del sistema sigue un mismo estándar, en cambio que el 40% afirman que no lo siguen, se concluye que la mayor parte de encuestados observan que el tipo, tamaño de fuente, textos, color, imágenes y botones siguen un mismo estándar.

A continuación en la tabla 17-3 se detalla los resultados de la pregunta 4.

Tabla 17-3: Pregunta 4

Pregunta 4. ¿Con la utilización del sistema se han reducido la cantidad de errores encontrados en los procesos?		
	Valor	Porcentaje
SI	5	100%
NO	0	0%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 10-3 se muestra los datos obtenidos de la pregunta 4.

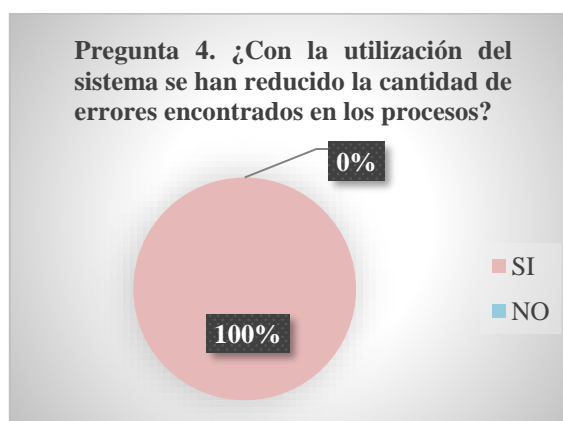


Figura 10-3. Resultados Pregunta 4

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 100% de las personas encuestadas opinaron que se han reducido considerablemente los errores encontrados en los procesos con la utilización del sistema “SGAF”.

A continuación en la tabla 18-3 se detalla los resultados de la pregunta 5.

Tabla 18-3: Pregunta 5

Pregunta 5. ¿Las acciones que solicita realizar el usuario en el sistema son fáciles de ejecutar?		
	Valor	Porcentaje
SI	5	100%
NO	0	0%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 11-3 se muestra los datos obtenidos de la pregunta 5.

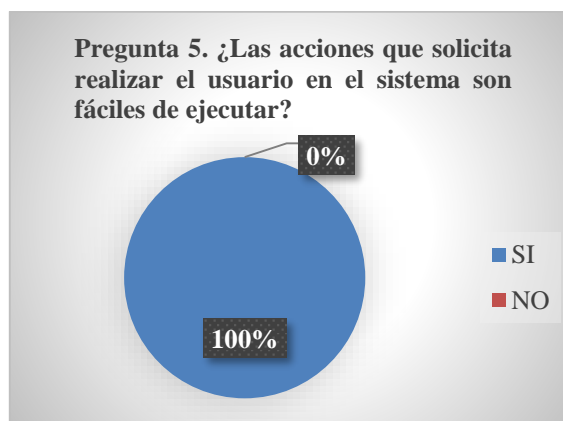


Figura 11-3. Resultados Pregunta 5
Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 100% de las personas encuestadas opinaron que las acciones que realizan son fáciles de ejecutar en el sistema.

A continuación en la tabla 19-3 se detalla los resultados de la pregunta 6.

Tabla 19-3: Pregunta 6

Pregunta 6. ¿Con la utilización del sistema se ha reducido el tiempo de respuesta en los procesos que se realizan?		
	Valor	Porcentaje
SI	5	100%
NO	0	0%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 12-3 se muestra los datos obtenidos de la pregunta 6.

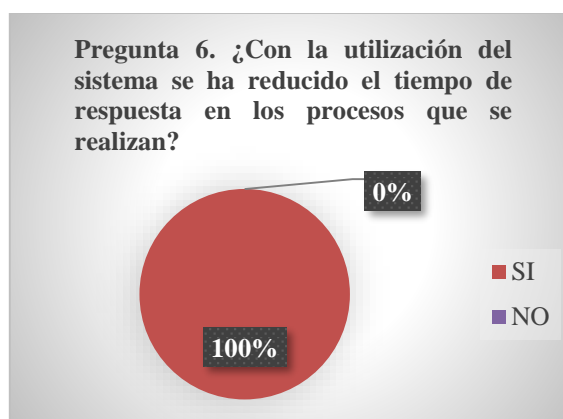


Figura 12-3. Resultados Pregunta 6
Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

El 100% de las personas encuestadas opinaron que al utilizar el sistema “SGAF” los tiempos en los cuales se realizan los procesos se han reducido considerablemente.

A continuación en la tabla 20-3 se detalla los resultados de la pregunta 7.

Tabla 20-3: Pregunta 7

Pregunta 7. ¿Es compatible el sistema con los diferentes navegadores?		
	Valor	Porcentaje
SI	5	100%
NO	0	0%
TOTAL	5	100%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 13-3 se muestra los datos obtenidos de la pregunta 7.



Figura 13-3. Resultados Pregunta 7

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

Análisis

Al 100% de las personas encuestadas opinaron que el sistema si se puede abrir en los diferentes navegadores.

Usabilidad del Sistema

La usabilidad es la capacidad del software de ser entendido, aprendido y usado en forma fácil. Cabe recalcar que algunos criterios de funcionalidad y eficiencia afectan la usabilidad, la misma que es determinada por los usuarios finales del software.

Los indicadores para medir la usabilidad del sistema “SGAF” se han elegido en base al estándar ISO 9126, el mismo que está orientada a la calidad del producto de software que lo clasifica en un conjunto estructurado de características. (Carreras, 2012)

De los criterios que se manejan en la usabilidad en base al estándar ISO 9126 se han tomado en cuenta los siguientes criterios para poder realizar el test de usabilidad del sistema “SGAF”, los mismos que se detallan a continuación:

Entendimiento.- es la capacidad que tiene el software para permitir al usuario entender si es adecuado, además de las ayudas que el software entrega. El indicador Guía del usuario están relacionadas con el criterio de entendimiento.

Operabilidad.- es la manera como el software permite al usuario ejecutarlo y controlarlo. Los indicadores de Accesibilidad y Compatibilidad están relacionadas con el criterio de operabilidad.

Aprendizaje: es la forma como el software permite al usuario aprender su uso. Los indicadores de Navegación y Contenido están relacionadas con el criterio de aprendizaje.

Eficiencia: es la forma del desempeño adecuado, de acuerdo al número de recursos utilizados según las condiciones planteadas. Este indicador esta relacionado con los tiempos de respuesta.

Eficacia: es el grado en que se realizan las actividades planificadas y se alcanzan los resultados planificados. Este indicador esta relacionado con la cantidad de errores encontrados.

A continuación en la tabla 21-3 se muestra el test para evaluar la usabilidad del sistema “SGAF”:

3.1.2.1 Validación de la usabilidad del sistema “SGAF”

Tabla 21-3: Test de Usabilidad

TEST DE USABILIDAD		SI	SI (%)	NO	NO (%)
	Indicador				
¿Existe un manual de ayuda que explique cómo utilizar el sistema “SGAF”?	Guía del usuario	5	100%	0	0%
¿El recorrido que se hace por el contenido del sistema es fácil?	Navegación	4	80%	1	20%
¿La presentación del contenido de la aplicación web como el tipo, tamaño de fuente, textos, color, imágenes y botones siguen el mismo estándar?	Contenido	3	60%	2	40%
¿Con la utilización del sistema se han reducido la cantidad de errores encontrados en los procesos?	Eficacia	5	100%	0	0%
¿Las acciones que solicita realizar el usuario en el sistema son fáciles de ejecutar?	Accesibilidad	5	100%	0	0%

¿Con la utilización del sistema se ha reducido el tiempo de respuesta en los procesos que se realizan?	Eficiencia	5	100%	0	0%
¿Es compatible el sistema con los diferentes navegadores?	Compatibilidad	5	100%	0	0%

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

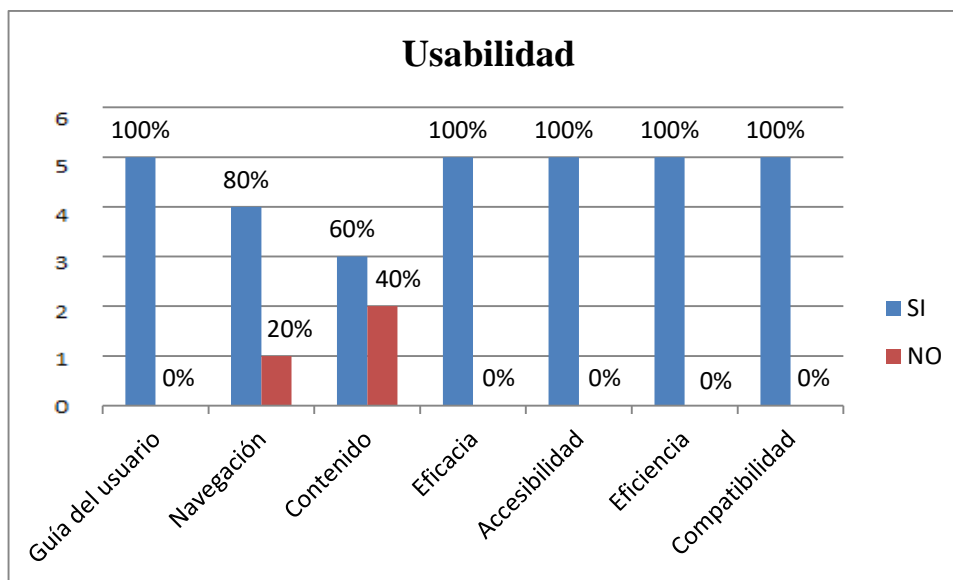


Figura 14-3. Resultados Usabilidad

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

En la Figura 14-3 se puede visualizar los resultados obtenidos acerca de la usabilidad del sistema “SGAF”.

Para realizar el análisis de los resultados de la encuesta se ha tomado en cuenta los valores porcentuales de las 7 preguntas de la “Tabla 21-3” de la columna SI(%), se realizó el promedio entre ellos obteniendo como resultado el 91.43%, podemos concluir que el sistema de Gestión Académica y Financiera “SGAF” es aceptable para los directivos y empleados de la institución.

3.2 Análisis de Resultados

De la tabla 8-3 con respecto al análisis comparativo de la toma de tiempos se obtienen los siguientes resultados:

En el proceso de inscripciones el tiempo empleado sin el sistema fue de 6 minutos con 29 segundos, mientras que con el uso del sistema es de 3 minutos con 10 segundos, existiendo entre ellos una diferencia de 3 minutos con 19 segundos.

En el proceso de consulta para conocer el valor que adeuda un estudiante el tiempo empleado sin el sistema fue de 1 minuto con 40 segundos, mientras que con el uso del sistema es de 48 segundos, existiendo entre ellos una diferencia de 52 segundos.

En el proceso de cobros a los estudiantes y generación de recibos, para la medición de los tiempos se tomó como muestra a 5 estudiantes, el tiempo empleado sin el sistema fue de 10 minutos con 44 segundos, mientras que con el sistema es de 2 minutos con 23 segundos, existiendo entre ellos una diferencia de 8 minutos con 21 segundos.

En los procesos relacionados para poder realizar el pago de un docente, el tiempo empleado sin el sistema fue de 26 minutos con 57 segundos, mientras que con el sistema es de 3 minutos con 44 segundos, existiendo entre ellos una diferencia de 23 minutos con 13 segundos.

En el proceso de generación del reporte para la contabilidad mensual, el tiempo empleado sin el sistema fue de 8 minutos con 43 segundos, mientras que con el sistema es de 1 minutos con 8 segundos, existiendo entre ellos una diferencia de 7 minutos con 35 segundos.

En relación a la toma de tiempos podemos concluir que con la utilización del sistema el tiempo en la realización de todos los procesos se redujo considerablemente.

De la tabla 9-3 y la tabla 12-3 con respecto a la cantidad de errores encontrados se obtienen los siguientes resultados:

El total de procesos que se realizan son 12, de los cuales en 9 de ellos existen errores lo que corresponde al 75%, el total de errores encontrados sin el uso del sistema son 21.

La cantidad de errores encontrados con la utilización del sistema fue de 5 errores, los mismos que fueron solucionados al momento de detectarlos.

CONCLUSIONES

- En el Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando” se realizan diferentes procesos tanto académicos como financieros los mismos que son registrados y manejados manualmente en cuadernos y hojas de Excel, la pérdida de tiempo, los errores encontrados y la falta de información oportuna provoca insatisfacción de los empleados y directivos.
- El framework JPA permite mapear las tablas a entidades, facilitando la conexión con la base de datos, además puede ser utilizada para la creación de cualquier tipo de aplicación, minimizando el tiempo de desarrollo del sistema. Además Primefaces es un framework open source que facilita el desarrollo de aplicaciones web ya que ofrece un gran conjunto de componentes que permiten crear interfaces que sean amigables, entendibles y fáciles de utilizar.
- Con la implementación del sistema de Gestión Académica y Financiera “SGAF” en relación al tiempo empleado en la realización de los principales procesos este se redujo en un total de 43 minutos con 20 segundos, puesto que sin el sistema se demoraban 54 minutos con 33 segundos y con el sistema 11 minutos con 13 segundos, razón por la cual se llega a la conclusión que el sistema “SGAF” constituye un gran aporte a la institución ya que emplea tiempos de respuesta menores a los utilizados manualmente.
- Los errores encontrados que correspondían al 75% de los procesos que se llevaban manualmente, fueron solucionados con la implementación del sistema “SGAF”, por lo que se concluye que el sistema redujo en un 100% los errores encontrados en los procesos.
- Una vez realizado los análisis correspondientes el sistema de Gestión Académica y Financiera “SGAF” logró el cumplimiento adecuado de todos los requerimientos solicitados por los directivos, por lo que se concluye que es aceptable en un 91.43%

RECOMENDACIONES

- Es necesario realizar un estudio previo del problema que se presenta en una empresa u organización para conocer cuáles son sus necesidades y así ofrecer la solución adecuada basada en un sistema automatizado.
- Los frameworks a utilizar deben ser investigados con anterioridad para poder conocer los aspectos más importantes para que al emplearlos no exista dificultades y así aprovechar al máximo las ventajas que ofrecen.
- Se debe utilizar la metodología Scrum ya que facilita la gestión de proyectos de cualquier tamaño y complejidad en los que los requerimientos están en constantes cambios, puesto que es fácil de implantar y los resultados que consigue son rápidos.
- Es fundamental conocer como funciona el framework JPA ya que simplifica la programación de la lógica lo que permite centrar los esfuerzos en optimizar consultas que realmente lo merecen, debido a que el código generado es más sencillo y mantenible. Además se debe tener en cuenta que algún cambio producido en la base de datos provocará que las tablas sean mapeadas nuevamente.
- A partir del sistema “SGAF” en base a la información almacenada en la base de datos se podría incorporar un nuevo sistema orientado a la educación virtual (e-learning).

GLOSARIO

Ajax.- es un conjunto de métodos y técnicas que permiten intercambiar datos con un servidor y actualizar partes de páginas web sin necesidad de recargar la página completamente.

Artefactos.- es cualquier obra manual realizada con un propósito o función técnica específica.

Anotaciones.- es una especie de comentario o meta dato que puede adjuntarse al código Java. Estas pueden ser procesadas durante las fases de compilación o durante la ejecución.

Aplicación web.- es cualquier aplicación que es accedida vía web por una red como internet e intranet.

Ciclo de Vida.- son las diferentes fases o partes que determinan como funciona un objeto o elemento.

Controlador.- es el encargado de reaccionar ante la petición del cliente, ejecuta la acción adecuada y crea el modelo pertinente.

Componente.- es aquello que forma parte de la composición de un todo.

CSS.- es un lenguaje utilizado en la presentación de documentos HTML, sirven para organizar la presentación y aspecto de una página web.

MVC.- es una patrón de arquitectura de software → (Modelo Vista Controlador).

Persistencia.- es la propiedad de los datos para subsistir de una manera u otra.

Roles.- son los cargos que desempeñan aquellas personas que forman parte de un proyecto.

Servlets.- son módulos escritos en Java que se utilizan en un servidor para extender sus capacidades de respuesta a los clientes.

BIBLIOGRAFÍA

- Alicante. (2015). Modelo vista controlador (MVC). Recuperado 23 de diciembre de 2015, a partir de <http://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
- Alicante, U. (2014, junio 26). Introducción a JavaServer Faces. Recuperado 13 de diciembre de 2015, a partir de <http://www.jtech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html#Caracter%C3%ADsticas+de+JSF>
- Alvarez, M. (2014, enero 2). Qué es MVC. Recuperado 28 de noviembre de 2015, a partir de <http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- Argenis. (2013, agosto 25). Fases del ciclo de vida de un componente JSF, ejemplos de immediate, y bypassUpdates | argenis' blog. Recuperado 15 de enero de 2016, a partir de <https://argenis85.wordpress.com/2011/08/25/fases-del-ciclo-de-vida-de-un-componente-jsf-ejemplos-de-immediate-y-bypassupdates/>
- Bahit, E. (2011, septiembre 14). Los roles en Scrum. Recuperado 13 de marzo de 2016, a partir de <http://www.desarrolloweb.com/articulos/roles-scrum.html>
- Caules, C. Á. (2013a, septiembre 17). Ejemplo de JPA , Introducción (I). Recuperado a partir de <http://www.arquitecturajava.com/ejemplo-de-jpa/>
- Caules, C. Á. (2013b, octubre 4). JPA (III) EntityManager métodos. Recuperado a partir de <http://www.arquitecturajava.com/jpa-entitymanager-metodos/>
- Chavarria, R. (2013, septiembre 29). Entendiendo el ciclo de vida de entidades JPA - R. Chavarria's Blog. Recuperado 2 de marzo de 2016, a partir de <http://rchavarria.github.io/blog/2013/09/29/entendiendo-ciclo-vida-entidades-jpa/>
- Climent. (2013, noviembre 13). carlescliment/curso-symfony2. Recuperado 18 de noviembre de 2015, a partir de <https://github.com/carlescliment/curso-symfony2>
- Cprime. (2012). What is AGILE? | What is SCRUM? | Agile FAQ's | cPrime. Recuperado 24 de marzo de 2016, a partir de <https://www.cprime.com/resources/what-is-agile-what-is-scrum/>
- Cruz, G. (2012, diciembre). *Primefaces*. Recuperado a partir de http://es.slideshare.net/gus_farfan/primefaces-14115155
- EcuRed. (2015). Arquitectura Cliente Servidor - EcuRed. Recuperado 5 de abril de 2016, a partir de http://www.ecured.cu/index.php/Arquitectura_Cliente_Servidor
- EcuRed. (2016, marzo 20). IReport - EcuRed. Recuperado 28 de marzo de 2016, a partir de <http://www.ecured.cu/index.php/IReport>
- Escobar, D. M. (2011, abril 7). Introducción a EJB 3.1 (IV) - davidmarco.es. Recuperado 10 de marzo de 2016, a partir de <http://www.davidmarco.es/articulo/introduccion-a-ejb-3-1-iv>
- Gómez, E. (2013, septiembre 19). JPA. JQL Java Persistence Query Language |. Recuperado a partir de <http://tic-tek.es/jpa-jql-java-persistence-query-language/>
- González, G. (2013, octubre 13). JavaServer Faces (JSF). Recuperado a partir de <https://kalistog.wordpress.com/jaserver-faces-jsf/>

- Jaf, R. (2013, julio 17). Uncategorized | ricardojaf7 [Primefaces]. Recuperado a partir de <https://ricardojaf7.wordpress.com/category/uncategorized/>
- Llanos, R., & Villa, O. (2011, marzo). *Exposicion JSF*. Recuperado a partir de <http://es.slideshare.net/castlallanos/exposicion-jsf>
- López, D. (2015, marzo 2). DISEÑO DE INVESTIGACIÓN. Recuperado 5 de febrero de 2016, a partir de <https://prezi.com/9yhvwc90unul/disenio-de-investigacion/>
- Lozu. (2013, julio 7). PATRONES DE DISEÑO: MODELO VISTA CONTROLADOR. Recuperado a partir de <http://thelozu.blogspot.com/2013/07/modelo-vista-controlador.html>
- Mancero, R. (2013, octubre 2). Sobre PostgreSQL | www.postgresql.org.es. Recuperado 12 de febrero de 2016, a partir de http://www.postgresql.org.es/sobre_postgresql
- Mojarra. (2015). PrimeFaces Showcase. Recuperado 5 de febrero de 2016, a partir de <http://www.primefaces.org/showcase/>
- Mosto, F. (2011, abril). *Scrum*. Recuperado a partir de <http://es.slideshare.net/mostofreddy/scrum-3624317>
- Moycas. (2011, mayo 4). ¿Qué es PostgreSQL? | Microbuffer. Recuperado 19 de noviembre de 2015, a partir de <https://microbuffer.wordpress.com/2011/05/04/que-es-postgresql/>
- Muñoz, J. M. (2012, julio 30). JSF, características principales, ventajas y puntos a destacar. Recuperado a partir de <https://josemmsimo.wordpress.com/2012/07/30/jsf-caracteristicas-principales-ventajas-y-puntos-a-destacar/>
- Peña, S. (2013, septiembre 1). JSF (Java Server Faces). Recuperado 5 de diciembre de 2015, a partir de <https://prezi.com/owge6wnmmi8v/jsf-java-server-faces/>
- Pincay, J. I. (2013, febrero). *Java y Jdbc*. Recuperado a partir de <http://es.slideshare.net/jpincay/java-y-jdbc>
- Rengifo, H. (2011, noviembre). *Jpa*. Recuperado a partir de <http://es.slideshare.net/hernanvid/jpa-10294017>
- Rojas, M. R. (2011, marzo 1). Ventajas y desventajas de las aplicaciones web : Vida MRR, blog de diseño web. Recuperado a partir de <http://www.vidamrr.com/2011/03/ventajas-y-desventajas-de-las.html>
- Romero, Y., & Díaz, Y. (2012, abril 1). Patrón Modelo - Vista - Controlador. Recuperado 1 de diciembre de 2015, a partir de <http://revistatelematica.cujae.edu.cu/index.php/tele/article/viewFile/15/10>
- Silva, R. (2012, mayo 1). Apresentação «SCRUM para Gerência de Projetos Prof. MSc. Robson Soares Silva www.robsonsoares.com Maio de 2012 III ERI MS.» Recuperado 27 de marzo de 2016, a partir de <http://slideplayer.com.br/slide/293289/>
- TuProgramación. (2013). ¿Qué es un ORM? Recuperado 9 de enero de 2016, a partir de <http://www.tuprogramacion.com/glosario/que-es-un-orm/>
- Tutorialspoint. (2016). JDBC Tutorial. Recuperado 12 de febrero de 2016, a partir de <http://www.tutorialspoint.com/jdbc/>

Uywork. (2015). Concepto de aplicación - Definición y Concepto. Recuperado 21 de noviembre de 2011, a partir de <http://concepto.de/aplicacion/>

Zulay, K. (2011, julio). *diseño de investigacion*. Recuperado a partir de <http://es.slideshare.net/alkemi/diseo-de-investigacion-8341669>

ANEXOS

ANEXO A: Técnicas de Recopilación de Información

Revisión de Documentos

GENIOS 1

DISP.	AULAS	CAPACIDAD	3-5	5-7	SABADOS	CAP.	PUPITRES MESAS
18	A1	22	CURSO NUEVO	AULA 1 (18)	A 1	(19)	Mover Aula 4 => 4 Pupitre ✓
14	A2	16	CURSO NUEVO	CURSO NUEVO	A 2	(15)	Mover Aula 4 => 2 Pupitres ✓
12	A3	12			A 3	(12)	OK
	A4						

GENIOS

GENIOS 2

DISP.	AULAS	CAPACIDAD	8-10	3-5	5-7	SABADOS	CAP.	PUPITRES MESAS
15	A9	16		AULA 10 (16)	AULA 2 + A13 (13)	A9	(11)	Mover Aula 14 => 1 Pupitre
22	A10	23		AULA 11 (18)	CURSO NUEVO	A10	(21)	Aumentar 1 silla
25	A11	23		AULA 9 + A 13 (24)	AULA 11 (17)	A11	(21)	Sacar 1 silla
27	A12	27	AULA 10 + A 11 (22)	AULA 12 (27)	AULA 12 (26)	A12	(21)	OK
13	A13	13		AULA 2 (13)	AULA 9 (13)	A13	(11)	OK
18	A14	18		AULA 14 (16)	AULA 10 (18)	A14	(16)	SACAR 1 Pupitre

GENIOS 3

DISP.	AULAS	CAPACIDAD	3-5	5-7	SABADOS	CAP.	PUPITRES MESAS
18	A5	22	Aula 17 (22)	Aula 17 (22)	Aula 8 + A 15	(19)	Mover A4 => 4 Pupitres
20	A6	25	CURSO NUEVO	CURSO NUEVO	Aula 6	(21)	Aumentar 5 sillas
21	A7	22	AULA 8 + A15 (22)	Aula 16 (19)	Aula 7	(13)	Aumentar 1 silla
18	A8	18	AULA 7 (14)	Aula 8 (17)	Aula 5	(12)	OK
16	A15	17	AULA 1 (17)	Aula 7 (16)	Aula 4	(12)	Aumentar 1 pupitre A4
34	A16	36	AULA 5 + A6 (36)	Aula 6 A14 (32)	Aula 16	(30)	Aumentar 2 Sillas
22	A17	25	AULA 16 (24)	Aula 5 a15 (25)	Aula 17	(21)	Aumentar 3 sillas

Figura 1-A. Información Aulas

Fuente: Institución “Genios Trabajando”

CURSOS SENESCYT (octubre - Marzo)							
N°	Fecha inicio	Fecha Fin	Día	Horario	Aula	Docente	Costo
1	✓ 30/11	31/03	La V.	8-10 AM	10	ING. DARIAN CARRERA	\$ 330
2	✓ 9/11	31/03	La V.	8-10 AM	11	ING. DIEGO SANDOVAL	\$ 360
3	✓ 19/10	31/03	La V.	3-5	1	ING. JUDY TORALUIS	\$ 330
	✓ 19/10	31/3	La V.	3-5	10	FRANKLIN V.	400
	✓ 19/10	31/3	La V.	3-5	11	SANDRO	400
	✓ 19/10	31/3	La V.	3-5	12	ING. FAUSTO H	330
	✓ 26/10	11	La V.	3-5	8	ANGEL BARRERO	400
	✓ 4/11	11	La V.	3-5	2	EDUARDO B.	(380)
	✓ 4/11	11	La V.	3-5	5	DANIEL C.	380
	✓ 16/11	11	La V.	3-5	15	ROSELY GONZALEZ	360
	✓ 30/11	11	La V.	3-5	9	DAVID JURADO	330
	✓ 14/12	11	La V.	3-5	6	ANDRES GUERRAS	300
	✓ 4/01	11	La V.	3-5	7	JORGE VAUSTECHI	280
	✓ 11/1	11	La V.	3-5	14	CONVICTO	170
	✓ 11/1	11	La V.	3-5	13	MARCO RIVERA	280
	✓ 18/1	11	11 11	3-5	17	ALEX U. ACEXIS CH.	250
	✓ 25/1	11	11	11	16	KEVIN HUMERT. BYRON.	220

Figura 2-A. Información Cursos

Fuente: Institución “Genios Trabajando”

NÓMINA DE DOCENTE DE GENIOS TRABAJANDO						
Cédula	Nombres	Apellidos	Celular	Teléfono	Dirección	Capacitación Dictada (genios/senescyt)
06046678-0	Edwardo Javier	Jarama Frag	0987027026	2626338	Lisboa y Londres	Senescyt
060475622-1	JUAN ANGEL	BARRERA BARRERA	0995300854		C. SIMONEZ 28-06 Y AVACUJO	SENECYT
1600680753	JAVIER ALEXANDER	TOBARIAN CARDASO	0997734883	2306199	CESAR LEON HERRERA Y AV MILTON PEÑAS	SENECYT
0202131363	LUIS ALBERTO	MONTAÑA ROMERO	0982856623	2856361	AMBATO JULIO CASTILLO Y RUISEÑO SANCHEZ	SENECYT
0605177310	ANDRES MARCELO	GUANABÁ MORENO	0984346613	2900135	GUANO	SENECYT
160054401	DIEGO DARIO	SANTOUR ATIAJA	0984238123		PANAMERICANA SUR	SENECYT
0604107334	FRANKLIN ORLANDO	VIZUETE VIZUETA	0983128289	2376703	TARAJI Y CA 35	SENECYT
1723574800	DAMIAN ALEXANDER	CARRERA GUINOLUISA	0980018143		10 DE AGOSTO Y ESTOCOLMO	SENECYT
1604776274	RAFAEL ALEXANDER	GARCIA URQUIZO	0979387419	2878167	Ciudadela Juan Montalvo	SENECYT
0603798095	JORGE LUIS	VEGISTEGUI ESCOBAR	0995454801		BARRIO 11 DE NOVIEMBRE	SENECYT
0604608729	ALEX FABIAN	URRUTIA CAYO	0983054974		Ciudadela DE LOS MAESTROS	SENECYT
0604353974	ALEXIS LEONARDO	CHAMBERO PARRA	0999210015	2366503	Ciudadela BACAPAGO 12 A CUSA 3	SENECYT
0603621103	MOAB IGNACIO	ARIAS AVALOS	0982101112	2941973	ORIENTE BOGOTAS Y LOS CEDROS	SENECYT
1604473497	BYRON MARCELO	ZURITA MESA	0999968636	2872667	CAROL CORTES	SENECYT
1604453057	DAVID EFRAIN	JURADO SALGUEIRO	0999208817	2751130	SAN DIEGO	SENECYT
0604591047	LUIS EDUARDO	GARCIA ROBAINO	0987376200	2966717	GARCIA MORENO Y CORDOBA	SENECYT
0603938150	KEVIN RUBEN	GARCIA BRAVO	0982246165		Ciudadela LUIS A. MARTINEZ	SENECYT
0604317982	MARCO IBRAIM	RIVERA ENRIQUE	0945359613	2967531	LOS PINO	SENECYT
1803657818	DIEGO FERNANDO	MAYORGA PEREZ	099729389	2316038	Ciudadela JOSE MARTELL	SENECYT
0603462227	FAUSTO OCTAVIO	HIDROBO NINO	0984496128		Ciudadela DE LOS MUESTRAS CIVILES	SENECYT
150825003	HUMBERTO	CALIN CUCIAMA	0983054543		Ciudadela JUAN MONTALVO	SENECYT

Figura 3-A. Información Docentes

Fuente: Institución "Genios Trabajando"

CENTRO DE CAPACITACIÓN ACADÉMICA Y PREUNIVERSITARIO GENIOS TRABAJANDO

DOCENTE: ING. DANIAN GERRERA

AULA: 15

HORARIO: (8-21:30)

FECHA DE INICIO: 24/OCTUBRE

DIAS: SABADOS

(15 personas)

N	NOMBRE	COLEGIO	CELULAR	CONVENCIONAL	INS	PAGO	PAGO	PAGO	PAGO	TOTAL
1	IVAN VERGARA	ALUSI	0990929521		30%	110%	10 10 10	100%	10%	10
2	JULISS LONHUA	ALUSI	0989726478		30%	110%	10 10 10	100%	10%	10
3	SABEL FIALLO	ALUSI	0994055978	2930121	30%	110%	10 10 10	100%	10%	10
4	NATHALY HERNANDEZ	ALUSI	0987337865		30%	110%	10 10 10	100%	10%	10
5	TALIA QUIROZ	ALUSI	0989726478		30%	110%	10 10 10	100%	10%	10
6	ODALIS PARADA	ALUSI			30%	110%	10 10 10	100%	10%	10
7	Claudia Barragan	Alausi	0993988430	2931254	30%	110%	10 10 10	100%	10%	10
8	PAUL CHACASABUZY		0984604147		30%	110%	10 10 10	100%	10%	10
9	CRISTOPHER DÍAZ		0968911921		30%	110%	10 10 10	100%	10%	10
10	CARLOS MOROCHO	CHIRIBOGA	0979124194		30%	110%	10 10 10	100%	10%	10
11	KATHERIN JUNTAMAY	ALUSI		2330045	30%	110%	10 10 10	100%	10%	10
12	BRANDON HARO	CHIRIBOGA	0987197116		30%	110%	10 10 10	100%	10%	10
13	VERONICA AWLEMA	SALASINAS	0995881380		30%	110%	10 10 10	100%	10%	10
14	KENNY TIXI	ALUSI			30%	110%	10 10 10	100%	10%	10
15	Luis Riera	ALUSI		2348053	30%	110%	10 10 10	100%	10%	10
16	DANIELA ENCAJADA	CHIRIBOGA		2607843	30%	110%	10 10 10	100%	10%	10
17										
18										
19										
20										
21										
22										
23										
24										

05/530 100/430 b/c. c/c 165

Figura 4-A. Información Inscripciones y Cuotas

Fuente: Institución "Genios Trabajando"

NOMBRE DOCENTE: EDUARDO ①

OTAL	IVA	LIBROS		MES NOVIEMBRE	AMORTIZACION	TOTAL CURSO PROYECCION	IVA	LIBROS	PAGO TOTAL	*0.8	PAGO PROFESOR	PAGO MENSUAL
				190	410							
				200	105							
				1,105								
PAGO MES NOVIEMBRE			390		AMORTIZACION	815						
						215	-390		Amort (1)	425		
									Amort (2)	150		
		Amort (1)	PAGO MES	Amort (2)					SALDO	390		
-12h30		190	200	10					GUARDAR	965		
1h30-10h30		200	120	140								
		390	140	150								
			x 0.3									
340	220	360	400	1220	370	6080	130	400	4950	3960	1190	240
410	670	300		440		4560	550	300	3710	2968	890	180
				810								420
SALDO 390												

5 MESES. 300 225

00 MES DICIEMBRE 390-
140-
430-
950

Figura 5-A. Información Proyección y Cálculo Pago

Fuente: Institución "Genios Trabajando"

NOMBRE	DICIEMBRE GUARDAR	PAGO GUARDAR	SALDO GUARDAR	Amort 8-12h30	Amort 1h30-10h30	PAGO 3-5	PAGO 5-7	PAGO V-S	MES DICIEMBRE	PAGO GUARDAR DICIEMBRE	SALDO DICIEMBRE	TOTAL GUARDAR	PAGO DICIEMBRE
EDUARDO	965	460	505	340	600	315	345		1600	420	1180	1685	880
DANIEL	685	180	505	310		360	305	205	1180	715	465	970	895
JUAN-JUAN	100	110	0	260					260		260	250	110
RAFAEL G	270	130	140	300		150	285		735	595	140	280	725
JESÚS	175	160	15	160					160		160	175	160
HUMBER POLLO	190	130	60	180					180		180	240	130
DAVID	290	150	140	330		165	220		715	625	190	330	675
DANIEL D	900	815	85	480	315				795		795	880	815
DOCTOR									240			930	435
HOLAB	220	160	60	390		225	115		730	340	390	450	500
SANDO	760	0	760			665	730	445	1840	1110	730	1490	1110
ANEELENO	810	480	330	250	60	315	265		890	510	380	710	990
FRANKLIN	380	150	230	190		470	145		805	460	345	575	610
JAVIER									1560				1560
ANDRÉS G	210	140	70	190		395			585	315	270	340	455
									12,275				10,050

Figura 6-A. Información Pago a Docentes

Fuente: Institución "Genios Trabajando"

ANEXO B: Diccionario de Datos

Tabla 1 – B: Diccionario de Datos

TABLA	COLUMNA	DESCRIPCIÓN	TIPO_TAMAÑO
AULA PK: id_aula FK: fk_sucursal_aula	id_aula	Identificador del aula	Integer
	descripción_aula	Descripción del aula	Variable characters (7)
	capacidad	Capacidad del aula	Integer
	id_sucursal	Identificador de la sucursal	Integer
CALCULO PAGO PK: num_pago, id_aula, id_curso FK: fk_calculopago_aula fk_calculopago_curso fk_calculopago_docente	num_pago	Identificador del número de pago	Integer
	id_aula	Identificador del aula	Integer
	id_curso	Identificador del curso	Integer
	id_docente	Identificador del docente	Integer
	fecha_inicio	Fecha inicio del cálculo	Date
	fecha_fin	Fecha fin del cálculo	Date
	ingreso	Dinero que ingreso por los cobros	Double precision
	iva	Ingreso * 0.12	Double precision
	libro	Ingreso por el valor de los libros	Double precision
	total_netto	Diferencia entre ingreso-iva-libro	Double precision
	porcentaje	Es el total neto * porcentaje	Double precision
CALCULO PAGO GENIOS PK: id_calculopagogenios FK: fk_calculopagogenios_docente	id_calculopagogenios	Identificador del cálculo pago genios	Integer
	id_docente	Identificador del docente	Integer
	suma	Dinero recaudado en la semana	Double precision
	valor_pagar	Valor a pagar al docente	Double precision
	fecha_inicio	Fecha inicio del cálculo	Date
	fecha_fin	Fecha fin del cálculo	Date
	genios	Saldo para genios de la semana	Double precision
CAPACITACION PK: id_capacitacion	id_capacitacion	Identificador de la capacitación	Integer
	descripción_capacitacion	Descripción de la capacitación	Variable characters (50)

	costo	Costo de la capacitación	Double precision
CIUDAD PK: id_ciudad FK: fk_ciudad_provincia	id_ciudad	Identificador de la ciudad	Integer
	nombre_ciudad	Nombre de la ciudad	Variable characters (100)
	id_provincia	Identificador de la provincia	Integer
COBRO PK: id_numcobro FK: fk_cobro_inscripcion fk_cobro_tipocobro	id_numcobro	Identificador del número de cobro	Integer
	id_estudiante	Identificador del estudiante	Integer
	id_curso	Identificador del curso	Integer
	fecha_cobro	Fecha del cobro	Date
	cantidad	Valor que paga el estudiante	Double precision
	id_tipocobro	Identificador del tipo de cobro	Integer
	id_aula	Identificador del aula	Integer
	id_docente	Identificador del docente	Integer
CONTABILIDAD DIARIA PK: id_contabilidad FK: fk_contabilidaddiaria_curso	id_contabilidad	Identificador de la contabilidad	Integer
	id_curso	Identificador del curso	Integer
	fecha	Fecha que se realiza la contabilidad	Date
	suma	Dinero recaudado por día y por curso.	Double precision
CONTABILIDAD DIARIA GENERAL PK: id_contabilidadgeneral	id_contabilidadgeneral	Identificador de la contabilidad	Integer
	fecha	Fecha que se realiza la contabilidad	Date
	total_genios	Dinero recaudado por día de genios	Double precision
	total_inscripcion	Dinero recaudado por día de inscripciones	Double precision
	total_senescyt	Dinero recaudado por día de senescyt, ineval	Double precision
	total_general	Dinero total recaudado por día	Double precision
CONTABILIDAD MENSUAL PK: id_contabilidadmensual FK: fk_mes_contabilidadmensual	id_contabilidadmensual	Identificador de la contabilidad mensual	Integer
	Id_mes	Identificador del mes	Integer
	fecha_inicio	Fecha inicio que se realiza la contabilidad mensual	Date
	fecha_fin	Fecha fin que se realiza la contabilidad mensual	Date
	total_inscripciones	Dinero recaudado por mes de inscripciones	Double precision
	total_genios	Dinero recaudado por mes de genios	Double precision

	total_senescyt	Dinero recaudado por mes de senescyt, ineval	Double precision
	total_gastos	Total de gastos mensuales	Double precision
	total_directivos	Total de pagos a docente	Double precision
	total_pagos	Dinero Total de los ingresos – los gastos	Double precision
CURSO PK: id_curso FK: fk_capacitacion_curso fk_horario_curso fk_periodo_curso	id_curso	Identificador del curso	Integer
	descripción_curso	Descripción del curso	Variable characters (50)
	id_capacitacion	Identificador de la capacitación	Integer
	id_periodo	Identificador del periodo	Integer
	id_horario	Identificador del horario	Integer
	duración	Duración del curso	Integer
DOCENTE PK: id_docente FK: fk_tipo_docente	id_docente	Identificador del docente	Integer
	ci_docente	Cédula del docente	Characters (10)
	nombres_docente	Nombres del docente	Variable characters (100)
	apellidos_docente	Apellidos del docente	Variable characters (100)
	celular_docente	Celular del docente	Characters (10)
	teléfono_docente	Teléfono del docente	Variable characters (9)
	dirección_docente	Dirección del docente	Variable characters (100)
	id_tipo	Tipo de docente	Integer
ESTUDIANTE PK: id_estudiante FK: fk_institucion_estudiante	id_estudiante	Identificador del estudiante	Integer
	ci_estudiante	Cédula del estudiante	Characters (10)
	nombres_estudiante	Nombres del estudiante	Variable characters (50)
	apellidos_estudiante	Apellidos del estudiante	Variable characters (75)
	email_estudiante	Correo electrónico del estudiante	Variable characters (100)
	redsocal_estudiante	Red social del estudiante	Variable characters (100)
	celular_estudiante	Celular del estudiante	Characters (10)
	representante_estudiante	Nombre del representante	Variable characters (100)
	celular_representante	Celular del representante	Characters (10)
	telefono_representante	Teléfono del representante	Variable characters (9)

	id_institucion	Identificador de la institución	Integer
GASTOS PK: id_gasto	id_gasto	Identificador del gasto	Integer
	fecha_gasto	Fecha en la que se realizó el gasto	Date
	solicitante	Nombre de la persona que solicita	Variable characters (70)
	descripción_gastos	Descripción del gasto realizado	Variable characters (50)
	valor	Dinero utilizado para el gasto	Double precision
USUARIO PK: id_usuario FK: fk_sucursal_usuario	id_usuario	Identificador del Usuario	Integer
	usuario	Nombre del usuario	Variable characters (100)
	clave	Contraseña	Variable characters (20)
	rol	Rol del usuario	Variable characters (50)
	cedula_usuario	Cédula del usuario	Variable characters (10)
	id_sucursal	Identificador de la Sucursal	Integer
TIPOHORARIO Pk: id_tipohorario	id_tipohorario	Identificador del tipo horario	Integer
	descripcion_tipohorario	Descripción del tipo de horario	Variable characters (50)
TIPOCOBRO Pk: id_tipocobro	id_tipocobro	Identificador del tipo cobro	Integer
	descripcion_tipocobro	Descripción del tipo de cobro	Variable characters (50)
TIPO Pk: id_tipo	id_tipo	Identificador del tipo	Integer
	descripcion_tipo	Descripción del tipo	Variable characters (50)
SUMINISTRO Pk: id_suministro	id_suministro	Identificador del suministro	Integer
	descripcion_suministro	Descripción del suministro	Variable characters (50)
	precio	Precio del suministro	Double precision
SUCURSAL Pk: id_sucursal FK: fk_ciudad_sucursal	id_sucursal	Identificador de la sucursal	Integer
	nombre_sucursal	Nombre de la sucursal	Variable characters (100)
	direccion_sucursal	Dirección de la sucursal	Variable characters (75)
	barrio	Barrio al que pertenece la sucursal	Variable characters (50)
	telefono_sucursal	Teléfono de la sucursal	Variable characters (9)
	correo_sucursal	Correo de la sucursal	Variable characters (50)
	id_ciudad	Ciudad de la sucursal	Integer

RECIBO Pk: num_recibo FK: fk_docente_recibo	num_recibo	Identificador de la sucursal	Integer
	Id_docente	Código docente	Integer
	fecha_inicio	Fecha inicio	Date
	fecha_fin	Fecha fin	Date
	fecha_recibo	Fecha del recibo	Date
	docente	Nombre del docente	Variable characters (100)
	saldo_guardado	Saldo guardado	Double precision
	total_cursos	Total de los cursos	Double precision
	pago_estimadosemana	Suma del pago estimado semana	Double precision
	pago_estimadofinsemana	Suma del pago estimado fin de semana	Double precision
	saldo_mes	Saldo sobrante del mes	Double precision
	total_guardado	Total del saldo guardado	Double precision
	pago_profesor	Valor a pagar al profesor	Double precision
PROYECCIONPAGO PK: id_aula, id_curso FK: fk_curso_proyeccion fk_aula_proyeccion fk_docente_proyeccion	id_aula	Aula	Integer
	id_curso	Curso	Integer
	id_docente	Docente	Integer
	fecha_calculo	Fecha de la proyección	Date
	ingreso	Ingreso total del curso	Double precision
	iva	Iva	Double precision
	libro	Total de libros	Double precision
	total_netto	El valor neto	Double precision
	porcentaje	Es un porcentaje de pago	Double precision
	amortizado	Es un porcentaje de error	Double precision
	pago_estimado	Pago estimado al docente	Double precision
	genios_directiva	Ganancia aproximada de la empresa	Double precision
PROVINCIA PK: id_provincia	id_provincia	Identificador de la provincia	Integer
	nombre_provincia	Nombre de la provincia	Variable characters (100)
PERÍODO	id_periodo	Identificador del período	Integer

PK: id_periodo FK: fk_tipo_periodo	fecha_inicio	Fecha inicio	Date
	fecha_fin	Fecha fin	Date
	id_tipo	Tipo de periodo	Integer
MES PK: id_mes	id_mes	Identificador del mes	Integer
	descripción_mes	Mes	Variable characters (25)
INSTITUCIÓN PK: id_institucion	id_institucion	Identificador de la institución	Integer
	nombre_institucion	Nombre de la institución	Variable characters (50)
	direccion_institucion	Dirección de la institución	Variable characters (70)
	telefono_institucion	Teléfono de la institución	Variable characters (9)
INSCRIPCIÓN PK: id_aula, id_curso, id_aula, id_docente FK: fk_curso_inscripcion fk_aula_inscripcion fk_docente_inscripcion fk_estudiante_inscripcion	id_aula	Aula	Integer
	id_curso	Curso	Integer
	id_docente	Docente	Integer
	id_estudiante	Estudiante	Integer
	fecha_inscripcion	Fecha de inscripcion	Date
	descuento	Descuento del curso	Double precision
HORARIO PK: id_horario FK: fk_tipo_horario fk_tipohorario_horario	id_horario	Identificador del horario	Integer
	dia	Días	Variable characters (50)
	hora	Horas	Variable characters (50)
	id_tipo	Tipo de horario	Integer
	id_tipohorario	Tipo de horario semana	Integer

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

ANEXO C: Historias de Usuario – Tareas de Ingeniería – Pruebas de Aceptación

Sprint 2

HU1: Registrar los datos de los estudiantes.

Tabla 1 – C: Historia de Usuario HU1

HISTORIA DE USUARIO	
Número: 1	Sprint Asignado: 2
Tiempo Estimado:	10 horas
Responsable:	Mercy Sani
Descripción: Como director requiero ingresar los datos de los estudiantes.	
Tarea 1: Diseño de la interfaz de usuario para el ingreso de los datos del estudiante <ul style="list-style-type: none"> - Diseñar una página para realizar el ingreso del estudiante. - Generar botones tanto para el Guardar como para el Cancelar. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla estudiante - Validación de los datos antes de ser guardarlos. 	

Realizado por: Mercy Sani

Tabla 2 – C: Tarea de Ingeniería 1 HU1

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Registrar los datos de los estudiantes	
Nombre de la Tarea: Diseño de la interfaz de usuario para el ingreso de los datos del estudiante.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita realizar el registro de los datos del estudiante.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla estudiante. 	

Realizado por: Mercy Sani

Tabla 3 – C: Prueba de Aceptación 1 HU1

PRUEBA DE ACEPTACIÓN	
Código: HU1.1	Nombre Historia de Usuario: Registrar los datos de los estudiantes
Nombre de la Prueba: Ingreso de los datos en la tabla estudiante	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de registrar los datos de los estudiantes es necesario verificar si fueron ingresados todos los campos en el formato correcto y almacenados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El estudiante de cédula '0604055568', no debe estar registrado en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el formulario con los datos que son obligatorios como es el caso de la cédula, nombres, apellidos, institución y los que no son como email, red social, celular, representante, celular representante, teléfono representante del estudiante cuya cédula es '0604055568' - Dar clic en el botón Guardar 	

Resultado Esperado: <ul style="list-style-type: none"> - Si existen campos obligatorios que no ha sido llenados, muestra un mensaje indicando que los llene. - Ingreso correcto del estudiante y emite el mensaje ‘Ingreso Correcto’
Evaluación de la Prueba: Satisfactoria.

Realizado por: Mercy Sani

Tabla 4 – C: Prueba de Aceptación 2 HU1

PRUEBA DE ACEPTACIÓN	
Código: HU1.2	Nombre Historia de Usuario: Registrar los datos de los estudiantes
Nombre de la Prueba: Validación de los datos antes de ser guardarlos.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario realizar la validación de los campos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Tener la validación correspondiente para cada campo. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el campo de la cédula del estudiante con caracteres alfanuméricos y especiales. - Llenar el campo de teléfono con caracteres alfabéticos. - Dar clic en el botón Guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Muestra mensaje de error de validación, porque el campo cédula acepta solo números al igual que el campo teléfono. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

HU5: Ingresar datos de los cursos (Capacitación/Nivelación).

Tabla 5 – C: Historia de Usuario HU5

HISTORIA DE USUARIO	
Número: 2	Sprint Asignado: 2
Tiempo Estimado:	16 horas
Responsable:	Mercy Sani
Descripción: Como director requiero ingresar los datos de los cursos	
Tarea 1: Creación de las funciones/métodos que sean necesarios para registrar los datos del curso. <ul style="list-style-type: none"> - Implementar el método para cargar los horarios dependiendo del tipo de curso. - Implementar el método para cargar los periodos dependiendo del tipo de curso. - Implementar el método para cargar las capacitaciones dependiendo del tipo de curso. 	
Tarea 2: Diseño de la interfaz de usuario para el ingreso de los datos del Curso. <ul style="list-style-type: none"> - Diseñar una página para realizar el ingreso del curso tanto para los cursos de Capacitación y Nivelación. - Generar botones tanto para el Guardar como para el Cancelar 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla curso - Verificación del código existente en la base de datos. 	

Realizado por: Mercy Sani

Tabla 6 – C: Tarea de Ingeniería 1 HU5

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Registrar los datos de los cursos	
Nombre de la Tarea: Creación de las funciones/métodos para registrar los datos del curso.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Creación de la función y la implementación del método requerido para poder realizar el ingreso de los datos del curso.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos que permiten cargar datos requeridos y de esta forma realizar el ingreso de los cursos. 	

Realizado por: Mercy Sani

Tabla 7 – C: Tarea de Ingeniería 2 HU5

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 2
Nombre Historia de Usuario: Registrar los datos de los cursos	
Nombre de la Tarea: Diseño de la interfaz de usuario para el ingreso de los datos del curso.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita realizar el registro de los datos del curso.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla curso. 	

Realizado por: Mery Sani

Tabla 8 – C: Prueba de Aceptación 1 HU5

PRUEBA DE ACEPTACIÓN	
Código: HU5.1	Nombre Historia de Usuario: Registrar los datos de los cursos
Nombre de la Prueba: Verificar el funcionamiento de los métodos que permiten cargar datos requeridos y de esta forma realizar el ingreso de los cursos.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar los métodos es necesario que se compruebe el funcionamiento adecuado de dicho proceso y de esta forma poder realizar el registro de cursos sin ningún inconveniente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a ser cargados deben estar almacenados en la base de datos para que puedan mostrarse, tomaremos como ejemplo el método “findHorarioEntities2 ()”. - Sintaxis correcta para implementar la función. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a los métodos por ejemplo findHorarioEntities2 (). - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar un listado de horarios dependiendo del query, en este caso muestra los horarios de tipo “Genios”. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 9 – C: Prueba de Aceptación 2 HU5

PRUEBA DE ACEPTACIÓN	
Código: HU5.2	Nombre Historia de Usuario: Registrar los datos de los cursos.
Nombre de la Prueba: Ingreso de los datos en la tabla curso	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de registrar los datos de los cursos es necesario verificar si fueron ingresados todos los campos en el formato correcto.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a registrar deben ser validados y controlados para que puedan ser almacenados en la base de datos. - El curso de descripción “Ineval”, horario “Lunes- Viernes 8:00 a 10:00 am”, período “24-04-2016”, no debe estar registrado en la base de datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el formulario con los datos que son especialmente obligatorios como es el caso de la descripción, capacitación, horario, período, duración, del curso cuyos datos son descripción “Ineval”, horario “Lunes- Viernes 8:00 a 10:00 am”, período “24-04-2016”. - Dar clic en el botón Guardar. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si existen campos obligatorios que no ha sido llenados, muestra un mensaje indicando que los llene. - Ingreso correcto del curso y emite el mensaje ‘Ingreso Correcto’ 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

HU10: Registrar nuevos usuarios al sistema.

Tabla 10 – C: Historia de Usuario HU10

HISTORIA DE USUARIO	
Número: 10	Sprint Asignado: 2
Tiempo Estimado:	6 horas
Responsable:	Mercy Sani
Descripción: Como director requiero la creación de nuevos usuarios.	
Tarea 1: Diseño de la interfaz de usuario para el ingreso de los datos del usuario <ul style="list-style-type: none"> - Diseñar una página para realizar el ingreso del estudiante - Generar botones tanto para el Guardar como para el Cancelar 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla usuario - Verificación de la cédula existente en la base de datos. 	

Realizado por: Mercy Sani

Tabla 11 – C: Tarea de Ingeniería 1 HU10

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 10
Nombre Historia de Usuario: Registrar los datos de los usuarios	

Nombre de la Tarea: Diseño de la interfaz de usuario para el ingreso de los datos del usuario.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita realizar el registro de los datos del usuario.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla usuario. 	

Realizado por: Mercy Sani

Tabla 12 – C: Prueba de Aceptación 1 HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.1	Nombre Historia de Usuario: Registrar los datos de los usuarios
Nombre de la Prueba: Ingreso de los datos en la tabla usuario	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de registrar los datos de los usuarios es necesario verificar si fueron ingresados todos los campos en el formato correcto y almacenados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El usuario de cédula ‘1803657848’, no debe estar registrado en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el formulario con los datos que son especialmente obligatorios como es el caso de la cédula, usuario, clave, rol y sucursal del usuario cuya cédula es ‘1803657848’ - Dar clic en el botón Guardar. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si existen campos obligatorios que no ha sido llenados, muestra un mensaje indicando que los llene. - Ingreso correcto del usuario y emite el mensaje ‘ Ingreso Correcto’. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 13 – C: Prueba de Aceptación 2 HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.2	Nombre Historia de Usuario: Registrar los datos de los usuario.
Nombre de la Prueba: Validación de los datos antes de guardarlos.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario realizar la validación de los campos que tienen los datos del usuario.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Tener la validación correspondiente para cada campo. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el campo de la cédula del usuario con caracteres alfanuméricos y especiales. - Dar clic en el botón Guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Muestra mensaje de error de validación, porque el campo cédula acepta solo números. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

HU12: Asignar el docente y el aula a los estudiantes que ya están inscritos.

Tabla 14 – C: Historia de Usuario HU12

HISTORIA DE USUARIO		
Número: 12		Sprint Asignado: 2
Tiempo Estimado:	20 horas	
Responsable:	Patricia Allauca	
Descripción: Como director requiero asignar el docente y el aula a los estudiantes que ya están inscritos.		
Tarea 1: Creación de la función e implementación del método que permita realizar la asignación del docente y aula <ul style="list-style-type: none">- Implementar la función f_asignar_aula_docente, el mismo que necesita 4 parámetros de entrada que son: estudiante, curso, docente y aula.- Actualizar los datos en la tabla inscripción y automáticamente se modificarán en la tabla cobro.- Implementar el método denominado actualizarInscripcion().		
Tarea 2: Diseño de la interfaz de usuario para realizar la asignación. <ul style="list-style-type: none">- Diseñar la página para buscar a los estudiantes, dado un curso.- Método que permita visualizar el listado de estudiantes de la base de datos denominado cargarInscripcionesxCurso()- Diseñar un diálogo que permita realizar la asignación del docente y el aula.- Generar botones tanto para el Guardar como para el Cancelar.		
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar la función que permita realizar la asignación.- Verificación de campos nulos y campos obligatorios.		

Realizado por: Patricia Allauca

Tabla 15 – C: Tarea de Ingeniería 1 HU12

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Asignar el docente y el aula a los estudiantes que ya están inscritos	
Nombre de la Tarea: Creación de la función e implementación del método que permita realizar la asignación del docente y aula.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Creación de la función que permita asignar el docente y el aula a los estudiantes inscritos.	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar la función que permita realizar la asignación.	

Realizado por: Patricia Allauca

Tabla 16 – C: Tarea de Ingeniería 2 HU12

TAREA DE INGENIERÍA	
Sprint: 2	Número de Tarea: 2
Nombre Historia de Usuario: Asignar el docente y el aula a los estudiantes que ya están inscritos.	
Nombre de la Tarea: Diseño de la interfaz de usuario para realizar la asignación .	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Diseñar la interfaz adecuada para realizar la asignación.	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificación de campos nulos y campos obligatorios.	

Realizado por: Patricia Allauca

Tabla 17 – C: Prueba de Aceptación 1 HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.1	Nombre Historia de Usuario: Asignar el docente y el aula a los estudiantes que ya están inscritos.
Nombre de la Prueba: Verificar la función que permita realizar la asignación.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar la función es necesario que se compruebe el funcionamiento adecuado para poder realizar la asignación.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El estudiante→8, el curso→39, docente→0 y aula→0 debe estar almacenados en la base de datos para poder asignar el docente y el aula. - Los estudiantes que no estén asignados a un docente y una aula deben tener el valor de 0 en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a la función f_asignar_aula_docente() e ingresar los datos: estudiante → 8, curso →39, docente →13 y aula →10 - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando el ingreso sea correcto y mostrar f→ cuando no se hayan registrado los datos. - Los datos se actualizaron en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 18 – C: Prueba de Aceptación 2 HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.2	Nombre Historia de Usuario: Asignar el docente y el aula a los estudiantes que ya están inscritos
Nombre de la Prueba: Verificación de campos nulos y campos obligatorios.	
Responsable: Ing. Diego Mayorga	
Descripción: Para realizar la asignación existen campos que hay que seleccionar como el código del docente y del aula.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Tener las validaciones correspondientes para los diferentes campos a registrar. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Seleccionar el docente→13 y el aula→10 para el estudiante→8 que está en el curso→39 - Ingresar dicha información para verificar las validaciones y así realizar la asignación correspondiente. - Almacenar la información. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si existen campos obligatorios que no han sido seleccionados, muestra un mensaje indicando que los llene. Por ejemplo seleccione el docente. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Sprint 3

HU19: Modificar los datos de los docentes

Tabla 19 – C: Historia de Usuario HU19

HISTORIA DE USUARIO		
Número: 19		Sprint Asignado: 3
Tiempo Estimado:	4 horas	
Responsable:	Mercy Sani	
Descripción: Como director requiero modificar los datos de los docentes.		
Actividades:		
Tarea 1: Diseño de la interfaz de usuario para la modificación de los datos del docente.		
<ul style="list-style-type: none">- Diseñar una página para realizar la modificación de los datos del docente.- Generar botones tanto para el Guardar como para el Cancelar.		
Pruebas de Aceptación:		
<ul style="list-style-type: none">- Mensaje de confirmación al momento de modificar los datos del docente.		

Realizado por: Mercy Sani

Tabla 20 – C: Tarea de Ingeniería 1 HU19

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Modificar los datos de los docentes	
Nombre de la Tarea: Diseño de la interfaz de usuario para la modificación de los datos del docente.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita realizar la modificación de los datos del docente.	
Pruebas de Aceptación:	
<ul style="list-style-type: none">- Mensaje de confirmación al momento de modificar los datos del docente.	

Realizado por: Mercy Sani

Tabla 21 – C: Prueba de Aceptación 1 HU19

PRUEBA DE ACEPTACIÓN	
Código: HU19.1	Nombre Historia de Usuario: Modificar los datos de los docentes
Nombre de la Prueba: Mensaje de confirmación al momento de modificar los datos del docente.	
Responsable: Ing. Diego Mayorga.	
Descripción: Al momento de modificar los datos de los docentes mostrar un mensaje de confirmación.	
Condiciones de Ejecución:	
<ul style="list-style-type: none">- Los datos del docente con cédula '0604867580' deben estar almacenados en la base de datos.- Validaciones de los datos que se van a ingresar.	
Pasos de Ejecución:	
<ul style="list-style-type: none">- Buscar en el listado de docentes al docente cuya cédula es '0604867580', seleccionarlo.- Dar clic en el botón Editar- Se muestra el mensaje correspondiente.	
Resultado Esperado:	
<ul style="list-style-type: none">- Los datos modificados correctamente son almacenados en la tabla docente la base de datos.- Al guardar la información modificada deberá mostrar un mensaje de confirmación o de error.	

Evaluación de la Prueba:
Satisfactoria.
Realizado por: Mercy Sani

HU26: Realizar el cambio de curso de los estudiantes.

Tabla 22 – C: Historia de Usuario HU26

HISTORIA DE USUARIO	
Número: 26	Sprint Asignado: 3
Tiempo Estimado:	40 horas
Responsable:	Patricia Allauca
Descripción: Como director requiero realizar el cambio de curso de los estudiantes.	
Tarea 1: Creación de la función e implementación del método que permita realizar el cambio de curso. <ul style="list-style-type: none"> - Obtener como parámetros los datos del estudiante, curso, docente y aula. - Calcular la capacidad disponible de los cursos. - Implementar la función f_cambio_curso. - Implementar la función f_capacidaddisponible. - Implementar el método denominado capacidadDisponible(). - Implementar el método denominado cambioCursoEstudiante(). - Modificar los datos en la tabla inscripción al realizar el cambio de curso. 	
Tarea 2: Diseño de la interfaz de usuario para realizar el cambio de curso de un estudiante. <ul style="list-style-type: none"> - Diseñar la página para buscar al estudiante por nombres o por la cédula. - Mostrar el listado de cursos disponibles. - Diseñar un diálogo para realizar el cambio de curso. - Generar botones tanto para el Guardar como para el Cancelar. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la función para realizar el cambio de curso. - Verificar la capacidad disponible de los cursos (aulas) a los que el estudiante puede cambiarse. - Mensaje de confirmación al realizar el cambio de curso. 	

Realizado por: Patricia Allauca

Tabla 23 – C: Tarea de Ingeniería 1 HU26

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Realizar el cambio de curso de los estudiantes	
Nombre de la Tarea: Creación de la función e implementación del método que permita realizar el cambio de curso.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Creación de la función que permita realizar el cambio de curso de los estudiantes.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la función para realizar el cambio de curso. 	

Realizado por: Patricia Allauca

Tabla 24 – C: Tarea de Ingeniería 2 HU26

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 2
Nombre Historia de Usuario: Realizar el cambio de curso de los estudiantes	
Nombre de la Tarea: Diseño de la interfaz de usuario	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Diseñar la interfaz de usuario que permita realizar el cambio de curso de los estudiantes	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la capacidad disponible de los cursos (aulas) a los que el estudiante puede cambiarse. 	

Realizado por: Patricia Allauca

Tabla 25 – C: Prueba de Aceptación 1 HU26

PRUEBA DE ACEPTACIÓN	
Código: HU26.1	Nombre Historia de Usuario: Realizar el cambio de curso de los estudiantes
Nombre de la Prueba: Verificar la función para realizar el cambio de curso.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar la función es necesario comprobar el funcionamiento para realizar el cambio de curso.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los cursos a los que el estudiante puede cambiarse deben estar almacenados en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Se desea cambiar al estudiante → 10, curso → 39, docente →13, aula →10 - Llamar a la función f_cambio_curso e ingresar los datos: estudiante→10, curso→11, docente→10 y aula→7 - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando el ingreso sea correcto y mostrar f→ cuando no se hayan registrado los datos. - Los datos se actualizaron en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 26 – C: Prueba de Aceptación 2 HU26

PRUEBA DE ACEPTACIÓN	
Código: HU26.2	Nombre Historia de Usuario: Realizar el cambio de curso de los estudiantes.
Nombre de la Prueba: Verificar la capacidad disponible de los cursos.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de verificar la capacidad disponible de los cursos es necesario realizar la búsqueda del estudiante y luego elegir el curso al que se podría cambiar.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Se podrán realizar el cambio siempre y cuando el curso tenga capacidad disponible. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Buscar al estudiante que desea realizar el cambio de curso. 	

<ul style="list-style-type: none"> - Generar un listado de todos los cursos posibles a los que el estudiante puede cambiarse. - Mostrar la capacidad disponible de los siguientes datos: curso→11, aula→7. - Ejecutar dicha instrucción.
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar la capacidad disponible en este caso 1.
Evaluación de la Prueba: Satisfactoria.
Realizado por: Patricia Allauca

Tabla 27 – C: Prueba de Aceptación 3 HU26

PRUEBA DE ACEPTACIÓN	
Código: HU26.3	Nombre Historia de Usuario: Realizar el cambio de curso de los estudiantes
Nombre de la Prueba: Mensaje de confirmación al realizar el cambio de curso	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de realizar el cambio de curso es necesario mostrar un mensaje de confirmación	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El nuevo curso→11 debe estar almacenado en la base de datos y tener capacidad disponible para poder realizar el cambio. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Escoger el nuevo curso→11, aula→7. - Dar clic en el botón Guardar. 	
Resultado Esperado: <ul style="list-style-type: none"> - Los datos son modificados correctamente en la base de datos. - Al guardar la información deberá mostrar un mensaje de confirmación “Datos Modificados correctamente”. - Cuando no se pueda registrar deberá mostrar un mensaje de error “no se puede registrar” 	
Evaluación de la Prueba: Satisfactoria.	
Realizado por: Patricia Allauca	

HU31: Listar los datos de los cursos actuales.

Tabla 28 – C: Historia de Usuario HU31

HISTORIA DE USUARIO	
Número: 31	Sprint Asignado: 3
Tiempo Estimado:	4 horas
Responsable:	Mercy Sani
Descripción: Como director requiero listar los datos de los cursos actuales.	
Actividades: Tarea 1: Diseño de la interfaz de usuario para generar el listado de los cursos actuales. <ul style="list-style-type: none"> - Diseñar una página para generar el listado de los cursos. - Método de visualización del listado de los cursos actuales. - Generar botones tanto para el Guardar como para el Cancelar. 	
Pruebas de Aceptación:	

<ul style="list-style-type: none"> - Visualización correcta de los datos de la tabla curso. - Mostrar un mensaje de aviso cuando no exista datos de los cursos.

Realizado por: Mercy Sani

Tabla 29 – C: Tarea de Ingeniería 1 HU31

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Listar los datos de los cursos actuales.	
Nombre de la Tarea: Diseño de la interfaz de usuario para generar el listado.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita mostrar el listado de los datos de los cursos.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Visualización correcta de los datos de la tabla curso. 	

Realizado por: Mercy Sani

Tabla 30 – C: Prueba de Aceptación 1 HU31

PRUEBA DE ACEPTACIÓN	
Código: HU31.1	Nombre Historia de Usuario: Listar los datos de los cursos actuales.
Nombre de la Prueba: Visualización correcta de los datos de la tabla curso.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario mostrar correctamente el listado de datos de los cursos actuales.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos de los cursos deben estar almacenados previamente en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Dar clic en el submenú Curso. - Visualizar el listado de los cursos. 	
Resultado Esperado: <ul style="list-style-type: none"> - Los datos de los cursos que se encuentran almacenados en la base de datos se mostrarán en la pantalla. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 31 – C: Prueba de Aceptación 2 HU31

PRUEBA DE ACEPTACIÓN	
Código: HU31.2	Nombre Historia de Usuario: Listar los datos de los cursos actuales.
Nombre de la Prueba: Mostrar un mensaje de aviso cuando no exista datos de los cursos	
Responsable: Ing. Diego Mayorga.	
Descripción: Es necesario mostrar un mensaje de aviso cuando no existan datos de los cursos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos de los cursos deben estar almacenados previamente en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Dar clic en el submenú Cursos. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si no existen datos de los cursos almacenados en la base de datos se mostrará el mensaje en la tabla que 	

diga “No existen Datos”.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Mercy Sani

HU40: Autenticación de usuarios para acceso al sistema.

Tabla 32 – C: Historia de Usuario HU40

HISTORIA DE USUARIO	
Número: 40	Sprint Asignado: 3
Tiempo Estimado:	40 horas
Responsable:	Mercy Sani
Descripción: Como director requiero la autenticación de usuarios para acceso al sistema.	
Tarea 1: Creación de la función/método que permita el inicio de sesión al sistema. <ul style="list-style-type: none"> - Implementar el método denominado Login en cual maneja la navegación por las páginas. - Implementar el método denominado Autenticar que verifica si el usuario existe o no. Tarea 2: Diseño de la interfaz de usuario para el inicio de sesión al sistema. <ul style="list-style-type: none"> - Diseñar una página para realizar el inicio de sesión. - Generar el botón Inicio de Sesión. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento del método que realiza la autenticación de usuarios para acceder al sistema. - Verificar el funcionamiento del botón Iniciar Sesión para realizar dicho proceso. 	

Realizado por: Mercy Sani

Tabla 33 – C: Tarea de Ingeniería 1 HU40

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Autenticación de usuarios para acceso al sistema.	
Nombre de la Tarea: Creación de la función/método que permita el inicio de sesión al sistema.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Creación de la función/método que permita el inicio de sesión de los usuarios para poder acceder al sistema.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos login y autenticar. 	

Realizado por: Mercy Sani

Tabla 34 – C: Tarea de Ingeniería 2 HU40

TAREA DE INGENIERÍA	
Sprint: 3	Número de Tarea: 2
Nombre Historia de Usuario: Autenticación de usuarios para acceso al sistema.	
Nombre de la Tarea: Diseño de la interfaz de usuario para el inicio de sesión al sistema.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar la interfaz de usuario que permita realizar el inicio de sesión de los usuarios para poder acceder al sistema.	
Pruebas de Aceptación:	

- Verificar el funcionamiento del botón Iniciar Sesión para realizar dicho proceso.

Realizado por: Mercy Sani

Tabla 35 – C: Prueba de Aceptación 1 HU40

PRUEBA DE ACEPTACIÓN	
Código: HU40.1	Nombre Historia de Usuario: Autenticación de usuarios para acceso al sistema.
Nombre de la Prueba: Verificar el funcionamiento de la función que realizan los métodos login y autenticar para el inicio de sesión al sistema.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de crear los métodos es necesario que se compruebe el funcionamiento adecuado de dicho proceso para poder realizar el logueo sin ningún inconveniente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El usuario cuyos datos son: cédula '1803657848' y clave 0, debe estar registrado en la base de datos. - Sintaxis correcta para implementar la función. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Ejecutar el query para la autenticación del usuario cuyos datos son: cédula '1803657848' y clave 0. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar datos en caso que existan. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 36 – C: Prueba de Aceptación 2 HU40

PRUEBA DE ACEPTACIÓN	
Código: HU40.2	Nombre Historia de Usuario: Autenticación de usuarios para acceso al sistema.
Nombre de la Prueba: Verificar el funcionamiento del botón Iniciar Sesión para realizar dicho proceso.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento del logueo, es necesario verificar si existen los datos ingresados en los campos y si están en el formato correcto.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a loguear deben ser validados y controlados para que puedan ser comparados en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el formulario con los datos que son especialmente obligatorios como es el caso del usuario y la contraseña del usuario como son: cédula '1803657848' y clave 0. - Dar clic en el botón Iniciar Sesión. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si son correctos los datos ingresados inicia sesión en el sistema. - Si no son correctos los datos muestra un mensaje 'Usuario no existe'. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Sprint 4

HU13: Registrar e imprimir los cobros de los estudiantes.

Tabla 37 – C: Historia de Usuario HU13

HISTORIA DE USUARIO	
Número: 13	Sprint Asignado: 4
Tiempo Estimado:	50 horas
Responsable:	Patricia Allauca
Descripción: Como director requiero registrar e imprimir los cobros de los estudiantes	
<p>Tarea 1: Creación de la función e implementación del método que permita registrar los cobros de las inscripciones de los estudiantes.</p> <ul style="list-style-type: none"> - Implementar la función f_insertar_cobro_cuota, el mismo que necesita 6 parámetros de entrada que son: estudiante, curso, docente, aula, fecha_cobro y cantidad. - Implementar el método denominado guardarCuotaSenescyt. - Insertar los datos en la tabla cobro. <p>Tarea 2: Diseño de la interfaz de usuario para realizar el registro de los cobros de los estudiantes.</p> <ul style="list-style-type: none"> - Diseñar la página para seleccionar el curso y mostrar los docentes y aulas en los que se dictan, posterior a ello buscar los estudiantes. - Método de visualización de los estudiantes de la base de datos denominado cargarAulasxCurso() - Diseñar un diálogo que permita realizar el cobro de las cuotas. - Generar botones tanto para el Guardar como para el Cancelar. <p>Tarea 3: Diseño de la interfaz de usuario y la generación del recibo utilizando JasperReport, convirtiéndola en formato pdf.</p> <ul style="list-style-type: none"> - Identificar los datos que se requieren para mostrar en el recibo del estudiante. - Implementar la función dentro del ireport para poder visualizar los datos de la inscripción del estudiante. - Implementar la función dentro del ireport para poder visualizar los datos de los pagos del estudiante. - Diseñar una página para el ingreso de los datos para poder generar el recibo de inscripción como son estudiante, curso y fecha. - Diseñar una página para el ingreso de los datos para poder generar el recibo de pago de los estudiantes. - Diseñar el formulario que permita realizar la conexión con la base de datos y que detalle los parámetros requeridos - Convertir el archivo en formato pdf 	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar la función para realizar el registro. - Verificar la generación del recibo de inscripción del estudiante en JasperReport. - Mostrar un archivo en formato pdf con el recibo de inscripción del estudiante. 	

Realizado por: Patricia Allauca

Tabla 38 – C: Tarea de Ingeniería 1 HU13

TAREA DE INGENIERÍA	
Sprint:4	Número de Tarea: 1
Nombre Historia de Usuario: Registrar e imprimir los cobros de los estudiantes.	
Nombre de la Tarea: Creación de la función e implementación del método que permita registrar los cobros de inscripción de los estudiantes.	

Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Creación de la función que permita registrar los cobros de la inscripción de los estudiantes.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la función para realizar el registro. 	

Realizado por: Patricia Allauca

Tabla 39 – C: Tarea de Ingeniería 2 HU13

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 2
Nombre Historia de Usuario: Registrar e imprimir los cobros de los estudiantes.	
Nombre de la Tarea: Diseño de la interfaz de usuario y la generación del recibo utilizando JasperReport, convirtiéndola en formato pdf.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Realizar la conexión con la base de datos para poder generar las instrucciones que permitan mostrar la información y generación del recibo del estudiante tanto para la inscripción como para el pago realizado utilizando JasperReport y convirtiéndola en formato pdf.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del recibo de inscripción del estudiante en JasperReport. 	

Realizado por: Patricia Allauca

Tabla 40 – C: Prueba de Aceptación 1 HU13

PRUEBA DE ACEPTACIÓN	
Código: HU13.1	Nombre Historia de Usuario: Registrar e imprimir los cobros de los estudiantes
Nombre de la Prueba: Verificar la función que permita realizar el registro de cobros de inscripción de los estudiantes.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar la función es necesario que se compruebe el funcionamiento para poder realizar el registro.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos de la inscripción de un estudiante deben estar almacenados en la base de datos para poder realizar los cobros. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a la función f_insertar_cobro_inscripcion e ingresar los datos estudiante→52, curso→38, docente→13, aula→10, fecha_cobro→'2015-10-05' y cantidad→30. - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando el ingreso sea correcto y mostrar f→ cuando no se hayan registrado los datos. - Los datos se han almacenado correctamente en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 41 – C: Prueba de Aceptación 2 HU13

PRUEBA DE ACEPTACIÓN

Código: HU13.2	Nombre Historia de Usuario: Registrar e imprimir los cobros de los estudiantes.
Nombre de la Prueba: Verificar la generación del recibo de inscripción del estudiante en JasperReport.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario realizar una conexión correcta con la base de datos para poder visualizar la información del recibo de inscripción del estudiante en JasperReport.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos del estudiante→52, curso→38, docente→13, aula→10, fecha_cobro→'2015-10-05' y cantidad→30 deben estar almacenados previamente para poder generar el recibo. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Crear la instrucción que devolverá los datos en el recibo. - Ingresar los parámetros requeridos para generar el recibo de inscripción del (estudiante→52, curso→38, docente→13, aula→10, fecha→'2015-10-05', cantidad→30). - Dar clic en OK. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando el ingreso sea correcto y mostrar f→ cuando no se hayan registrado los datos. - Los datos se han almacenado correctamente en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 42 – C: Prueba de Aceptación 3 HU13

PRUEBA DE ACEPTACIÓN	
Código: HU13.3	Nombre Historia de Usuario: Registrar e imprimir los cobros de los estudiantes.
Nombre de la Prueba: Mostrar un archivo en formato pdf con el recibo de inscripción del estudiante.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario que en el recibo se muestre los datos requeridos y así poder visualizarlos en pantalla en un archivo pdf.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El recibo debe estar generado previamente con el JasperReport. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Escoger el estudiante →52. - Escoger el curso→ 38. - Escoger al docente→ 13. - Escoger el aula→ 10. - Seleccionar la fecha→ '2015-10-05'. - Escribir la cantidad→ 30. - Dar clic en Mostrar Recibo. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar en una nueva página el recibo de inscripción del estudiante en un archivo pdf. - Imprimir el recibo. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

HU15: Realizar la proyección de los cursos.

Tabla 43 – C: Historia de Usuario HU15

HISTORIA DE USUARIO	
Número: 15	Sprint Asignado: 4
Tiempo Estimado:	30 horas
Responsable:	Mercy Sani
Descripción: Como director requiero realizar la proyección de los cursos.	
Tarea 1: Creación de los métodos que permita registrar los datos de la proyección. <ul style="list-style-type: none">- Crear las funciones f_insertar_calculo_pago y f_insertar_calculo_pago_genios en la base de datos.- Identificar los datos de la proyección para registrarlos en la base de datos.- Implementar el método para cargar los cursos actuales.- Implementar el método para mostrar los docentes y los respectivos cursos.- Implementar el método guardar proyección.- Insertar los datos en la tabla proyeccionpago.	
Tarea 2: Diseño de la interfaz de usuario para el ingreso de los datos de la proyección. <ul style="list-style-type: none">- Diseñar una página para cargar los cursos actuales.- Diseñar una página para mostrar los docentes del curso seleccionado.- Diseñar una página para realizar el ingreso de la proyección del curso y docente seleccionados.- Generar botones tanto para el Guardar como para el Cancelar	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar el funcionamiento de los métodos requeridos para poder realizar la proyección.- Ingreso de los datos en la tabla proyeccionpago.	

Realizado por: Mercy Sani

Tabla 44 – C: Tarea de Ingeniería 1 HU15

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Registrar los datos de las proyecciones	
Nombre de la Tarea: Creación de los métodos que permita registrar los datos de la proyección.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Creación e implementación de los métodos requeridos para poder realizar el ingreso de los datos de la proyección.	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar el funcionamiento de los métodos que nos permitirán obtener los datos para realizar el ingreso de las proyecciones en la base de datos.	

Realizado por: Mercy Sani

Tabla 45 – C: Tarea de Ingeniería 2 HU15

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 2
Nombre Historia de Usuario: Registrar los datos de los horarios	
Nombre de la Tarea: Diseño de la interfaz de usuario para el ingreso de los datos de la proyección.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani

Descripción: Diseñar las interfaces de usuario que permita realizar el registro de los datos de la proyección.
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla proyeccionpago

Realizado por: Mercy Sani

Tabla 46 – C: Prueba de Aceptación 1 HU15

PRUEBA DE ACEPTACIÓN	
Código: HU15.1	Nombre Historia de Usuario: Realizar la proyección de los cursos
Nombre de la Prueba: Verificar el funcionamiento de los métodos requeridos para poder realizar la proyección.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar los métodos es necesario que se compruebe el funcionamiento adecuado que permite cargar datos y de esta forma poder realizar el registro de las proyecciones sin ningún inconveniente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a ser cargados deben estar almacenados en la base de datos para que puedan mostrarse, tomaremos como ejemplo el método “mostrarDocenteCurso (26, 17, 12)”. - Sintaxis correcta para implementar la función. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a los métodos creados para realizar la proyección como por ejemplo mostrarDocenteCurso (id curso, id docente, id aula), con los siguientes datos curso 26, docente 12 y aula 17. - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar un listado de los docentes del curso al cual se le va a realizar la proyección. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 47 – C: Prueba de Aceptación 2 HU15

PRUEBA DE ACEPTACIÓN	
Código: HU15.2	Nombre Historia de Usuario: Realizar la proyección de los cursos.
Nombre de la Prueba: Ingreso de los datos en la tabla proyeccionpago	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de registrar los datos de las proyecciones es necesario verificar si fueron ingresados todos los campos en el formato correcto.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a registrar deben ser validados y controlados para que puedan ser almacenados en la base de datos. - La proyección con los siguientes datos: curso 26, aula 17, docente 12, porcentaje 0.3, amortizado 0.8, fecha ‘2015-10-31’ no deben estar registrado en la base de datos. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llenar el formulario con los datos que son obligatorios como es el caso del curso, aula, docente, porcentaje, amortizado, fecha de la proyección cuyos datos son: curso 26, aula 17, docente 12, porcentaje 0.3, amortizado 0.8 - Dar clic en el botón Guardar. 	

Resultado Esperado: <ul style="list-style-type: none"> - Si existen campos obligatorios que no ha sido llenados, muestra un mensaje indicando que los llene. - Ingreso correcto de la proyección y emite el mensaje 'Ingreso Correcto'.
Evaluación de la Prueba: Satisfactoria.
Realizado por: Mercy Sani

HU16: Calcular el pago de los docentes.

Tabla 48 – C: Historia de Usuario HU16

HISTORIA DE USUARIO		
Número: 16		Sprint Asignado: 4
Tiempo Estimado:	50 horas	
Responsable:	Patricia Allauca	
Descripción: Como director requiero calcular el pago de los docentes.		
Tarea 1: Creación de la función e implementación del método que permita realizar el cálculo del pago de los docentes. <ul style="list-style-type: none">- Identificar los datos del curso, docente, aula, fecha fin, porcentaje y el número de retirados requeridos para realizar el cálculo.- Implementar la función f_insertar_calculo_pago.- Seleccionar los datos requeridos.- Implementar el método guardarCalculo().		
Tarea 2: Diseño de la interfaz de usuario que permita realizar el cálculo. <ul style="list-style-type: none">- Diseñar una página que permita buscar los cursos.- Método para visualizar el listado de los docentes.- Diseñar una página para realizar el cálculo del pago de los docentes- Generar botones tanto para el Guardar como para el Cancelar.		
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar la función para realizar dicho proceso.- Ingreso de los datos en la tabla calculopago.- Cálculos generados correctamente.		

Realizado por: Patricia Allauca

Tabla 49 – C: Tarea de Ingeniería 1 HU16

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Como director requiero calcular el pago de los docentes.	
Nombre de la Tarea: Creación de la función e implementación del método que permita realizar el cálculo del pago de los docentes.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Implementar el método requerido para poder realizar el cálculo para el pago del docente.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la función para realizar dicho proceso. 	

Realizado por: Patricia Allauca

Tabla 50 – C: Tarea de Ingeniería 2 HU16

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 2
Nombre Historia de Usuario: Como director requiero calcular el pago de los docentes.	
Nombre de la Tarea: Diseño de la interfaz de usuario que permita realizar el cálculo.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Diseñar la interfaz de usuario que permita realizar el cálculo del pago de los docentes.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Ingreso de los datos en la tabla calculopago. 	

Realizado por: Patricia Allauca

Tabla 51 – C: Prueba de Aceptación 1 HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.1	Nombre Historia de Usuario: Calcular el pago de los docentes
Nombre de la Prueba: Verificar la función para realizar dicho proceso.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar la función es necesario que se compruebe el funcionamiento para poder realizar el cálculo correspondiente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los cursos deben estar iniciados para poder realizar el cálculo. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a la función e ingresar los datos: f_insertar_calculo_pago(curso→3, aula→1, docente→3, fechafin→‘2015-10-31’, porcentaje→0.3, retirado→0). - Ejecutar dicha instrucción. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando la función se haya ejecutado correctamente y mostrar f→ cuando no se hayan registrado los datos. - Los datos se almacenaron correctamente en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 52 – C: Prueba de Aceptación 2 HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.2	Nombre Historia de Usuario: Calcular el pago de los docentes
Nombre de la Prueba: Ingreso de los datos en la tabla calculo pago	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario verificar que los valores calculados junto con los demás campos se ingresen correctamente en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos deben estar almacenados previamente. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Seleccionar el curso para el que se desea realizar el cálculo por ejemplo curso→4. 	

<ul style="list-style-type: none"> - Mostrar los docentes que dictan este curso y seleccionar el docente → 19. - Ingresar la fecha fin es decir hasta que fecha→'2015-11-19'. - Ingresar el porcentaje de descuento→0.3 - Ingresar el número de estudiantes retirados→0 - Dar clic en el botón Guardar.
Resultado Esperado: <ul style="list-style-type: none"> - Datos almacenados correctamente en la tabla calculopago en la base de datos.
Evaluación de la Prueba: Satisfactoria.
Realizado por: Patricia Allauca

Tabla 53 – C: Prueba de Aceptación 3 HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.3	Nombre Historia de Usuario: Calcular el pago de los docentes
Nombre de la Prueba: Cálculos generados correctamente	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario comprobar que los valores calculados sean correctos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los porcentajes deben ser ingresados con el formato correcto. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Una vez ingresados los datos docente→19, curso→4, aula→14, fechafin → '2015-'10-31', porcentaje 0.3, retirados→0. 	
Resultado Esperado: <ul style="list-style-type: none"> - Los valores calculados son: ingreso→2120, iva→254.40, libro→550, totalneto→1315.60, porcentaje→394.68; los cuales coinciden con los datos calculados manualmente por el usuario y se almacenan correctamente en la base de datos. 	
Evaluación de la Prueba: Satisfactoria.	
Realizado por: Patricia Allauca	

HU27: Realizar la devolución de las cuotas si el estudiante se ha retirado.

Tabla 54 – C: Historia de Usuario HU27

HISTORIA DE USUARIO	
Número: 27	Sprint Asignado: 4
Tiempo Estimado:	30 horas
Responsable:	Mercy Sani
Descripción: Como director requiero realizar la devolución de las cuotas si el estudiante se ha retirado.	
Actividades: Tarea 1: Creación de los métodos que permita realizar la devolución de las cuotas del estudiante. <ul style="list-style-type: none"> - Identificar los datos que permitan realizar la devolución de las cuotas. - Implementar el método para cargar los estudiantes que han realizado pagos de los cursos actuales. - Implementar el método para cargar los cobros de un estudiante seleccionado. - Eliminar o realizar la devolución de los cobros del estudiante seleccionado. 	

Tarea 2: Diseño de la interfaz de usuario para la devolución de los cobros del estudiante.	
<ul style="list-style-type: none"> - Diseñar una página para cargar los estudiantes que han realizados pagos de los cursos actuales. - Diseñar una página que muestre los datos de los estudiantes que se hayan buscado por indicio de nombre. - Diseñar una página que muestres los cobros realizados al estudiante seleccionado. - Generar botones tanto para el Eliminar como para el Cancelar. 	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos requeridos para poder realizar la devolución. - Eliminación de los datos en la tabla cobro. 	
Realizado por: Mercy Sani	

Tabla 55 – C: Tarea de Ingeniería 1 HU27

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Realizar la devolución de las cuotas si el estudiante se ha retirado.	
Nombre de la Tarea: Creación de los métodos que permitan realizar la devolución de las cuotas del estudiante.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Creación e implementación de los métodos requeridos para poder realizar la devolución de cuotas pagadas por el estudiante.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos que permitirán obtener los datos para realizar la devolución de las cuotas pagadas por el estudiante. 	

Realizado por: Mercy Sani

Tabla 56 – C: Tarea de Ingeniería 2 HU27

TAREA DE INGENIERÍA	
Sprint: 4	Número de Tarea: 2
Nombre Historia de Usuario: Realizar la devolución de las cuotas si el estudiante se ha retirado.	
Nombre de la Tarea: Diseño de la interfaz de usuario para la devolución de los cobros del estudiante.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Diseñar las interfaces de usuario que permita realizar la devolución de las cuotas.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Eliminación de los datos en la tabla cobro. 	

Realizado por: Mercy Sani

Tabla 57 – C: Prueba de Aceptación 1 HU27

PRUEBA DE ACEPTACIÓN	
Código: HU27.1	Nombre Historia de Usuario: Realizar la devolución de las cuotas si el estudiante se ha retirado.
Nombre de la Prueba: Verificar el funcionamiento de los métodos que permitirán obtener los datos para realizar la devolución de las cuotas pagadas por el estudiante.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar los métodos es necesario que se compruebe el funcionamiento adecuado	

que permite cargar datos y de esta forma poder realizar la eliminación de las cuotas pagadas por el estudiante sin ningún inconveniente.
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos a ser cargados deben estar almacenados en la base de datos para que puedan mostrarse, por ejemplo el método cargarcobrosxEstudiante(id estudiante), siendo el id estudiante 62. - Sintaxis correcta para implementar los métodos.
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar al método cargarcobrosxEstudiante(62) creado para realizar la devolución. - Ejecutar dicha instrucción.
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar un listado de los cobros de las cuotas del estudiante para realizar la devolución.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Mercy Sani

Tabla 58 – C: Prueba de Aceptación 2 HU27

PRUEBA DE ACEPTACIÓN	
Código: HU27.2	Nombre Historia de Usuario: Realizar la devolución de las cuotas si el estudiante se ha retirado.
Nombre de la Prueba: Eliminación de los datos en la tabla cobro.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de eliminar los datos de los cobros del cual se realiza la devolución es necesario verificar que los datos fueron eliminados en la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El cobro cuyos datos son: estudiante 62, docente 13, curso 38, cantidad 110, debe estar en la base de datos para realizar su eliminación. - Los datos a eliminar deben ser validados y controlados para poder realizar el proceso sin ningún inconveniente. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Buscar al estudiante al cual se desea realizar la devolución en este caso de id 62. - Seleccionar el cobro de la cuota que se le va a devolver, con los siguientes datos: estudiante 62, docente 13, curso 38, cantidad 110. - Dar clic en el botón Eliminar. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si el cobro o cobros de las cuotas fueron eliminados, muestra el mensaje 'Datos Eliminados'. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Sprint 5

HU28: Generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.

Tabla 59 – C: Historia de Usuario HU28

HISTORIA DE USUARIO	
Número: 28	Sprint Asignado: 5

Tiempo Estimado:	16 horas
Responsable:	Patricia Allauca
Descripción: Como administrador requiero generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.	
Tarea 1: Diseño de la interfaz de usuario y la generación e impresión del reporte de los pagos efectuados por los estudiantes dado un curso utilizando JasperReport y convirtiéndola en formato pdf. <ul style="list-style-type: none"> - Mostrar un archivo en formato pdf con el reporte de los pagos efectuados por parte de los estudiantes. - Identificar los datos que se requieren para mostrar en el reporte. - Implementar la función dentro del ireport para poder visualizar los datos. - Diseñar una página para seleccionar el curso del cual se desea obtener los pagos efectuados por parte de los estudiantes. - Diseñar el dialogo que permita cargar automáticamente al curso e inmediatamente permita mostrar el reporte. - Convertir el archivo en formato pdf. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del reporte de los pagos efectuados por los estudiantes en JasperReport. - Mostrar un archivo en formato pdf con el reporte de los pagos efectuados por parte de los estudiantes. 	

Realizado por: Patricia Allauca

Tabla 60 – C: Tarea de Ingeniería 1 HU28

TAREA DE INGENIERÍA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.	
Nombre de la Tarea: Diseño de la interfaz de usuario y la generación e impresión del reporte de los pagos efectuados por los estudiantes dado un curso utilizando JasperReport y convirtiéndola en formato pdf.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Realizar la conexión con la base de datos para poder generar las instrucciones que permitan mostrar la información de los pagos efectuados de los estudiantes utilizando JasperReport y convirtiéndola en formato pdf.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del reporte de los pagos efectuados por los estudiantes en JasperReport. 	

Realizado por: Patricia Allauca

Tabla 61 – C: Prueba de Aceptación 1 HU28

PRUEBA DE ACEPTACIÓN	
Código: HU28.1	Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.
Nombre de la Prueba: Verificar la generación del reporte de los pagos efectuados por los estudiantes en JasperReport.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario realizar una conexión correcta con la base de datos para poder visualizar la información de los pagos efectuados por los estudiantes en JasperReport.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - En el curso→8 y aula→11 deben estar estudiantes inscritos para poder visualizar el reporte. 	

Pasos de Ejecución: <ul style="list-style-type: none"> - Crear la instrucción que devolverá los datos en el reporte de los pagos efectuados por los estudiantes. - Ingresar los parámetros requeridos: curso→8 y aula→11. - Dar clic en OK.
Resultado Esperado: <ul style="list-style-type: none"> - Si existen datos se mostrará en la pantalla el reporte de los pagos efectuados caso contrario emitirá un mensaje The document has no pages.
Evaluación de la Prueba: Satisfactoria.
Realizado por: Patricia Allauca

Tabla 62 – C: Prueba de Aceptación 2 HU28

PRUEBA DE ACEPTACIÓN	
Código: HU28.2	Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos efectuados por los estudiantes dado un curso.
Nombre de la Prueba: Mostrar un archivo en formato pdf con el reporte de los pagos efectuados de los estudiantes.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario que el reporte de los pagos efectuados por los estudiantes se muestre los datos requeridos y así poder visualizarlos en pantalla en un archivo pdf.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El recibo debe estar generado previamente con el JasperReport. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Escoger el curso→8 y el aula→11. - Dar clic en Mostrar Reporte. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar en una nueva página el reporte de los pagos efectuados por los estudiantes en un archivo pdf. - Imprimir el reporte. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

HU29: Generar el reporte de los estudiantes dado un curso.

Tabla 63 – C: Historia de Usuario HU29

HISTORIA DE USUARIO	
Número: 29	Sprint Asignado: 5
Tiempo Estimado:	10 horas.
Responsable:	Mercy Sani
Descripción: Como director requiero imprimir el reporte del listado de los estudiantes dado un curso.	
Tarea 1.- Diseño de la interfaz de usuario y la generación del listado utilizando JasperReport y convirtiéndola en formato pdf. <ul style="list-style-type: none"> - Identificar los datos que se requieren para mostrar en el listado de los estudiantes de un curso. - Implementar la función dentro del ireport para poder visualizar los datos. - Diseñar una página para el ingreso de los datos para poder generar el listado de estudiantes como son 	

<p>curso y aula.</p> <ul style="list-style-type: none"> - Diseñar el formulario que permita realizar la conexión con la base de datos, que detalle los parámetros requeridos. - Convertir el archivo en formato pdf.
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar la generación del listado en JasperReport. - Mostrar un archivo en formato pdf con el listado de los estudiantes de un curso.

Realizado por: Mercy Sani

Tabla 64 – C: Tarea de Ingeniería 1 HU29

TAREA DE INGENIERÍA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: Generar e imprimir el reporte de los estudiantes dado un curso.	
Nombre de la Tarea: Diseño de la interfaz de usuario y la generación del listado utilizando JasperReport y convirtiéndola en formato pdf	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Realizar la conexión con la base de datos para poder generar las instrucciones que permitan mostrar la información y generación del listado utilizando JasperReport y convirtiéndola en formato pdf.	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar la generación del listado en JasperReport. 	

Realizado por: Mercy Sani

Tabla 65 – C: Prueba de Aceptación 1 HU29

PRUEBA DE ACEPTACIÓN	
Código: HU29.1	Nombre Historia de Usuario: Generar e imprimir el reporte de los estudiantes dado un curso.
Nombre de la Prueba: Verificar la generación del listado en JasperReport	
Responsable: Ing. Diego Mayorga.	
Descripción: Es necesario realizar una conexión correcta con la base de datos para poder visualizar la información del listado en JasperReport.	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> - No debe existir una conexión entre JasperReport y la base de datos. - Los datos a mostrarse deben estar almacenados en la base de datos. 	
<p>Pasos de Ejecución:</p> <ul style="list-style-type: none"> - Crear la conexión desde JasperReport con la base de datos creada en PostgreSQL. - Crear la instrucción que devolverá los datos en el listado. - Ingresar el parámetro como el curso y aula. - Dar clic en OK. 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> - Si existen datos del listado de los estudiantes de un curso se mostrará dichos datos en la pantalla caso contrario emitirá un mensaje The document has no pages. 	
<p>Evaluación de la Prueba:</p> <p>Satisfactoria.</p>	

Realizado por: Mercy Sani

Tabla 66 – C: Prueba de Aceptación 2 HU29

PRUEBA DE ACEPTACIÓN	
Código: HU29.2	Nombre Historia de Usuario: Generar e imprimir el reporte de los estudiantes dado un curso.
Nombre de la Prueba: Mostrar un archivo en formato pdf con el listado de los estudiantes.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario que en el listado se muestre los datos requeridos y así poder visualizarlos en pantalla en un archivo pdf.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos requeridos deben estar almacenados en la base de datos se podrá imprimir el listado. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Escoger el curso ‘Senecyt , lunes- viernes 15:00 a 17:00’ y el aula ‘A09’. - Dar clic en Imprimir Listado. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar en una nueva página el listado en un archivo pdf. - Imprimir el listado. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

HU36: Emitir el reporte de los pagos a los docentes dada una fecha.

Tabla 67 – C: Historia de Usuario HU39

HISTORIA DE USUARIO	
Número: 39	Sprint Asignado: 5
Tiempo Estimado:	14 horas
Responsable:	Patricia Allauca
Descripción: Como administrador requiero generar e imprimir el reporte de los pagos a los docentes dada una fecha.	
Tarea 1: Diseño de la interfaz de usuario y la generación e impresión del reporte de los pagos a los docentes dada una fecha utilizando JasperReport y convirtiéndola en formato pdf. <ul style="list-style-type: none"> - Identificar los datos que se requieren para mostrar en el reporte. - Implementar la función dentro del ireport para poder visualizar los datos. - Diseñar una página para seleccionar la fecha que se desea mostrar el reporte de los pagos a los docentes. - Convertir el archivo en formato pdf. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del reporte de los pagos a los docentes dada una fecha. - Mostrar un archivo en formato pdf con el reporte de los pagos a los docentes, dada una fecha. 	

Realizado por: Patricia Allauca

Tabla 68 – C: Tarea de Ingeniería 1 HU39

TAREA DE INGENIERÍA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos a los docentes dada una fecha.	
Nombre de la Tarea: Diseño de la interfaz de usuario y la generación e impresión del reporte de los pagos a los	

docentes dada una fecha utilizando JasperReport y convirtiéndola en formato pdf.	
Tipo de Tarea: Desarrollo	Programador Responsable: Patricia Allauca
Descripción: Realizar la conexión con la base de datos para poder generar las instrucciones que permitan mostrar la información de los pagos de los docentes utilizando JasperReport y convertirla en formato pdf.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del reporte de los pagos a los docentes en JasperReport. 	

Realizado por: Patricia Allauca

Tabla 69 – C: Prueba de Aceptación 1 HU39

PRUEBA DE ACEPTACIÓN	
Código: HU39.1	Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos a los docentes dada una fecha.
Nombre de la Prueba: Verificar la generación del reporte de los pagos a los docentes en JasperReport.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario realizar una conexión correcta con la base de datos para poder visualizar la información de los pagos a los docentes en JasperReport.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Se debe ingresar una fecha en la que se haya realizado los pagos a los docentes para generar el reporte. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Crear la instrucción que devolverá los datos en el reporte de los pagos a los docentes. - Ingresar el parámetro de la fecha → '2015-10-31'. - Dar clic en OK. 	
Resultado Esperado: <ul style="list-style-type: none"> - Si existen datos se mostrará en la pantalla el reporte de los pagos a los docentes caso contrario emitirá un mensaje The document has no pages. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Patricia Allauca

Tabla 70 – C: Prueba de Aceptación 2 HU39

PRUEBA DE ACEPTACIÓN	
Código: HU39.2	Nombre Historia de Usuario: Generar e imprimir el reporte de los pagos a los docentes dada una fecha.
Nombre de la Prueba: Mostrar un archivo en formato pdf con el reporte de los pagos a los docentes.	
Responsable: Ing. Diego Mayorga	
Descripción: Es necesario que el reporte de los pagos a los docentes se muestre los datos requeridos y así poder visualizarlos en pantalla en un archivo pdf.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - El recibo debe estar generado previamente con el JasperReport. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Seleccionar la fecha '2015-10-31'. - Dar clic en Mostrar Reporte. 	

Resultado Esperado: <ul style="list-style-type: none"> - Diseñar una página para seleccionar la fecha que se desea mostrar el reporte de los pagos a los docentes. - Mostrar en una nueva página el reporte de los pagos de los docentes en un archivo pdf. - Imprimir el reporte.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Patricia Allauca

HU37: Registrar y emitir el reporte de la contabilidad diaria.

Tabla 71 – C: Historia de Usuario HU37

HISTORIA DE USUARIO	
Número: 37	Sprint Asignado: 5
Tiempo Estimado:	40 horas
Responsable:	Mercy Sani
Descripción: Como director requiero registrar e emitir el reporte de la contabilidad diaria.	
Tarea 1: Creación de la función/método que permita registrar los datos de la contabilidad diaria. <ul style="list-style-type: none"> - Identificar los datos de la contabilidad para registrarlos en la base de datos. - Implementar la función f_insertar_contabilidad_diaria. - Insertar los datos en la tabla contabilidaddiaria. - Implementar el método denominado guardarContabilidadDiaria(). 	
Tarea 2.- Diseño de la interfaz de usuario y la generación del reporte utilizando JasperReport y convirtiéndola en formato pdf. <ul style="list-style-type: none"> - Identificar los datos que se requieren para mostrar en el reporte de la contabilidad diaria. - Implementar la función dentro del ireport para poder visualizar los datos. - Diseñar una página para el ingreso de los datos para poder generar el reporte como es la fecha de la cual se desea emitir la contabilidad diaria. - Diseñar el formulario que permita realizar la conexión con la base de datos, que detalle los parámetros requeridos. - Convertir el archivo en formato pdf. 	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que el método implementado devuelva el reporte solicitado. - Verificar la generación del listado en JasperReport. - Mostrar un archivo en formato pdf con el listado de los estudiantes de un curso. 	

Realizado por: Mercy Sani

Tabla 72 – C: Tarea de Ingeniería 1 HU37

TAREA DE INGENIERÍA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: El sistema permitirá registrar y emitir el reporte de la contabilidad diaria.	
Nombre de la Tarea: Creación de la función/método que permita registrar los datos de la contabilidad diaria.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Creación de la función que permita registrar la contabilidad diaria.	
Pruebas de Aceptación:	

- Verificar que el método implementado devuelva el reporte solicitado.
--

Realizado por: Mercy Sani

Tabla 73 – C: Tarea de Ingeniería 2 HU37

TAREA DE INGENIERÍA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: El sistema permitirá registrar y emitir el reporte de la contabilidad diaria.	
Nombre de la Tarea: Diseño de la interfaz de usuario y la generación del reporte utilizando JasperReport y convirtiéndola en formato pdf.	
Tipo de Tarea: Desarrollo	Programador Responsable: Mercy Sani
Descripción: Realizar la conexión con la base de datos para poder generar las instrucciones que permitan mostrar la información del reporte utilizando JasperReport y convirtiéndola en formato pdf.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar la generación del listado en JasperReport. 	

Realizado por: Mercy Sani

Tabla 74 – C: Prueba de Aceptación 1 HU37

PRUEBA DE ACEPTACIÓN	
Código: HU37.1	Nombre Historia de Usuario: El sistema permitirá registrar y emitir el reporte de la contabilidad diaria.
Nombre de la Prueba: Verificar que el método implementado devuelva el reporte solicitado.	
Responsable: Ing. Diego Mayorga	
Descripción: Al momento de generar la función es necesario que se compruebe el buen funcionamiento de dicho proceso para poder realizar el registro sin ningún inconveniente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - La contabilidad diaria cuyos datos son: curso 3, fecha '2016-03-10' no deben estar registrados en la base de datos. - Sintaxis correcta para implementar la función. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Llamar a la función f_insertar_contabilidad_diaria: (id_curso,fecha), cuyos datos son curso 3, fecha '2016-03-10'. - Ejecutar dicha instrucción 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar t→ cuando el ingreso sea correcto y mostrar f→ cuando no se hayan registrado los datos. - Visualizar los datos que se han registrado. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Tabla 75 – C: Prueba de Aceptación 2 HU37

PRUEBA DE ACEPTACIÓN	
Código: HU37.2	Nombre Historia de Usuario: El sistema permitirá registrar y emitir el reporte de la contabilidad diaria.

Nombre de la Prueba: Verificar la generación del reporte en JasperReport.
Responsable: Ing. Diego Mayorga.
Descripción: Es necesario realizar una conexión correcta con la base de datos para poder visualizar la información del reporte en JasperReport.
Condiciones de Ejecución: <ul style="list-style-type: none"> - No debe existir una conexión entre JasperReport y la base de datos. - Los datos a mostrarse deben estar almacenados en la base de datos, para que se puedan visualizarse.
Pasos de Ejecución: <ul style="list-style-type: none"> - Crear la conexión desde JasperReport con la base de datos creada en PostgreSQL - Crear la instrucción que devolverá los datos en el reporte. - Ingresar el parámetro como el curso y fecha. - Dar clic en OK
Resultado Esperado: <ul style="list-style-type: none"> - Si existen datos de la contabilidad diaria se mostrará dichos datos en la pantalla caso contrario emitirá un mensaje The document has no pages.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Mercy Sani

Tabla 76 – C: Prueba de Aceptación 3 HU37

PRUEBA DE ACEPTACIÓN	
Código: HU37.3	Nombre Historia de Usuario: El sistema permitirá registrar y emitir el reporte de la contabilidad diaria.
Nombre de la Prueba: Mostrar un archivo en formato pdf con el reporte de la contabilidad diaria.	
Responsable: Ing. Diego Mayorga.	
Descripción: Es necesario que en el reporte se muestre los datos requeridos y así poder visualizarlos en pantalla en un archivo pdf.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Los datos deben estar almacenados en la base de datos se podrá imprimir el reporte. 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Escoger el curso 3 ‘Senecyt , lunes- viernes 15:00 a 17:00’ y la fecha‘ 2016-03-10’. - Dar clic en Mostrar Recibo. 	
Resultado Esperado: <ul style="list-style-type: none"> - Mostrar en una nueva página el reporte de la contabilidad diaria en un archivo pdf. - Imprimir el listado. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Mercy Sani

Planificación del Trabajo de Titulación

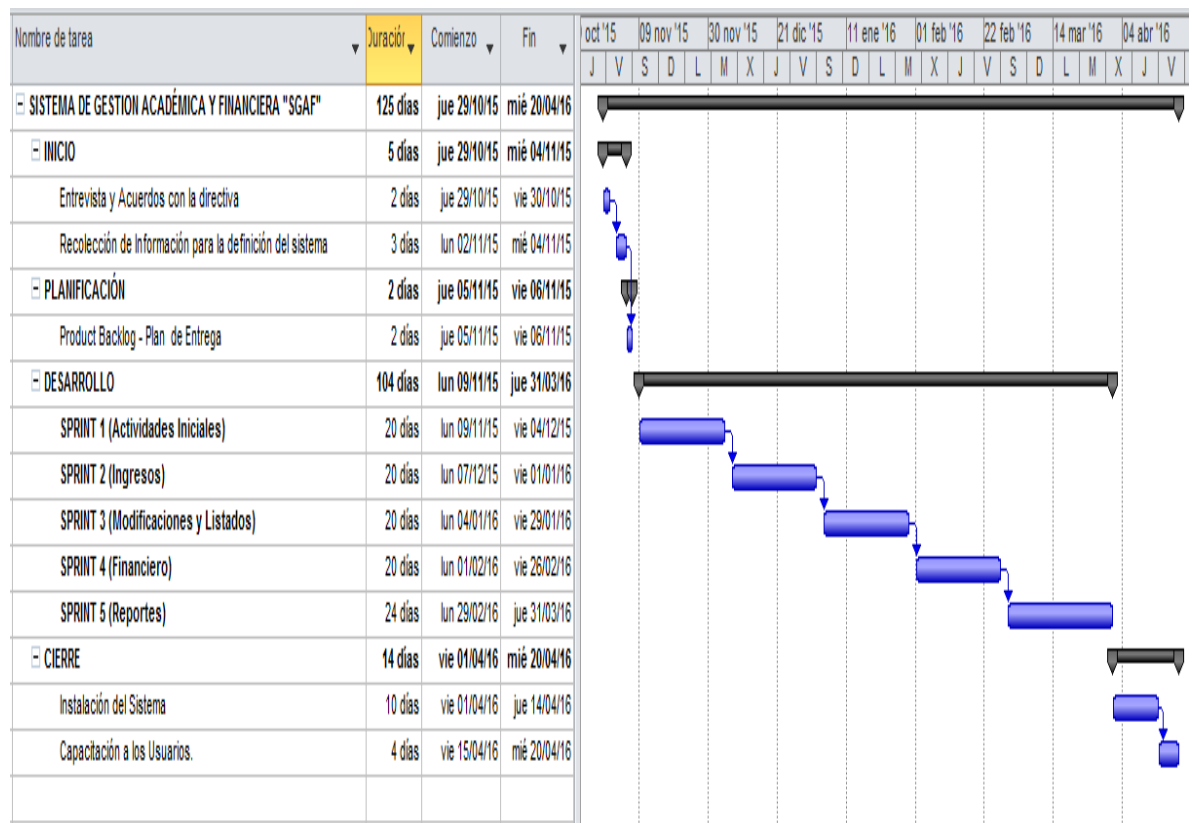


Figura 1-C. Planificación del Trabajo de Titulación “SGAF”

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

ANEXO D: Ficha de Observación con respecto al tiempo

Institución:

Tema:

Lugar:

Nombres del investigador(es):

Fuente:

Fecha de la observación:

FICHA DE OBSERVACIÓN (Tiempo)

Tabla 1-D: Ficha de Observación (Tiempo)

Procesos	Unidad de medida	Tiempo empleado

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

ANEXO E: Ficha de Observación con respecto a la cantidad de errores

Institución:

Tema:

Lugar:

Nombres del investigador(es):

Fuente:

Fecha de la observación:

FICHA DE OBSERVACIÓN (Cantidad de errores encontrados)

Tabla 1-E: Ficha de Observación (Cantidad de errores encontrados)

Procesos	Nº	Descripción del error	Observación

Realizado por: Allauca Oñate, Patricia; Sani Centeno, Mercy, 2016

ANEXO F: Encuesta para medir la usabilidad

La presente encuesta se la realiza para obtener información acerca de la usabilidad del sistema desarrollado en el Centro de Capacitación Académica y PreUniversitaria “Genios Trabajando”.

Por favor tómese unos minutos para completar la siguiente encuesta.

1. ¿Existe un manual de ayuda que explique cómo utilizar el sistema “SGAF”?

SI ☐

NO ☐

2. ¿El recorrido que se hace por el contenido del sistema es fácil?

SI ☐

NO ☐

3. ¿La presentación del contenido del sistema como el tipo, tamaño de fuente, textos, color, imágenes y botones siguen el mismo estándar?

SI ☐

NO ☐

4. ¿Con la utilización del sistema se han reducido la cantidad de errores encontrados en los procesos?

SI ☐

NO ☐

5. ¿Las acciones que solicita realizar el usuario en el sistema son fáciles de ejecutar?

SI ☐

NO ☐

6. ¿Con la utilización del sistema se ha reducido el tiempo de respuesta en los procesos que se realizan?

SI ☐

NO ☐

7. ¿Es compatible el sistema con los diferentes navegadores?

SI ☐

NO ☐